MINI PROJECT

REPORT ON

“AirFare”

Submitted By
“Chavhan Mayur”
“Nalbhe Narayan”
“Patil Bhagyashree”
“Patil Dhanashri”
“Rale Akhil”

Under the Guidance of
“Prof. Milind Deshkar”

Submitted To
Savitribai Phule Pune University

As a partial fulfillment for the award of
the degree of
MASTER OF COMPUTER APPLICATION
Semester: 1

At
ASM’s
Institute of Business Management and
Research, Chinchwad, Pune – 19

(Affiliated to SPPU & Approved by AICTE)

Session: 2021-23

INDEX

1 : INTRODUCTION

The “AirFare” has been developed to override the problems prevailing in practicing manual system. We have designed admin login, login and sign up forms using Java swing, through which one can enter into AirFare. We have also added forget tab, in case the assigned in person be able to recover credentials he/she registered at time of registration. Once assigned person created an account can login into the system, he/she can be able to access search flight, book ticket and cancel ticket.

He/ She may go to login form back and can go to home and exist window. The assigned person can search flight by given details as source, destination and date. So it will display the flight name with its price and timing. Then he/she can book ticket by using this flight details, by giving customer name and no of seats, after this he/she pay through debit and credit card, onces payment is done the ticket is booked successfully. Hence ticket is generated for the customer in print ticket, he/she has to only give ticket no and can see the entire ticket. As well as on the request of customer, the booked ticket can be cancelled by given ticket no and cancel date. Hence Cancelled ticket is generated in print cancel ticket.

The assigned person can also be able to see the entire booked ticket information in view ticket option. And even can see the entire information related to cancelled tickets in deleted ticket option. In admin login, admin can login by giving admin name and password. Onces Admin has login, can add flight details, go back to login form again and can exit window from home page. Add flight can be done by giving flight id, flight name, source, destination, date, arrival and departure time, flight price.

Admin can also search, update and delete the particular flight details. the entire AirFare project is designed and created in Java. All the details filled with Java program is stored in the database using MySQL.

1.1 Existing System and Need for System

Existing System:

The existing Airline reservation system was suffering from a series of drawbacks. Since whole of the system was to be maintained with hands the process of keeping, maintaining and retrieving the information was very tedious and lengthy. The records were never used to be in a systematic order. there used to be lots of difficulties in associating any particular transaction with a particular context.

If any information was to be found it was required to go through the different registers, documents there would never exist anything like report generation. There would always be unnecessary consumption of time while entering records and retrieving records. One more problem was that it was very difficult to find errors while entering the records. Once the records were entered it was very difficult to update these records.

The reason behind it is that there is lot of information to be maintained and have to be kept in mind while running this airline system. Due to the above drawback, the existing system is very complicated and not secure system.

Need for the System:

Now our system will overcome this all drawbacks. It will reduce efforts required to manage all airline records. All work can be done on just few clicks. The Only need is to fill given forms for retrieving required information. This system will provide Booking of Tickets and Cancel the booked tickets. Admin can maintain /modify /create flight details. user can be able to make payment through debit and credit card for Ticket. Faster technique to generate flight ticket. Better service and Minimum time required. Booking tickets and cancellations of tickets is possible from anywhere to any place. Even user can view the stored details of booked ticket and cancelled tickets.

1.2 Scope of Work

The section provide scope of the work. AirFare system is used by both admin and user. Our system is user-friendly.

1. Admin can login into the system can add modify update and delete the flight details.
2. User has to register into the system and then only can login and use the system. i.e only registered user can use our system.
3. If registered user forgot the credentials while login. Then they can recover the credentials from forget tab and then login to the system.
4. Registered User can search the detail of flights for the purpose of booking a ticket.
5. Registered User can be able to book ticket and make a payment with option debit card or credit card.
6. Registered User can cancel the booked ticket if he/she is willing to cancel.
7. Tickets are generated for booked ticket as well as cancelled ticket.

1.3 Operating Environment –

- CLIENT SIDE:

Operating System	Windows 10,11
Processor	I5
RAM	Greater than 4 GB
Browser	Internet Explorer, Google Chrome, etc.

- SERVER SIDE:

Operating system	Windows 10,11
Frontend	Java Swing
Middle ware	Java Development Kit (JDK 17), NetBeans
Backend	MySQL Database Server
Server	Apache Tomcat, Xampp

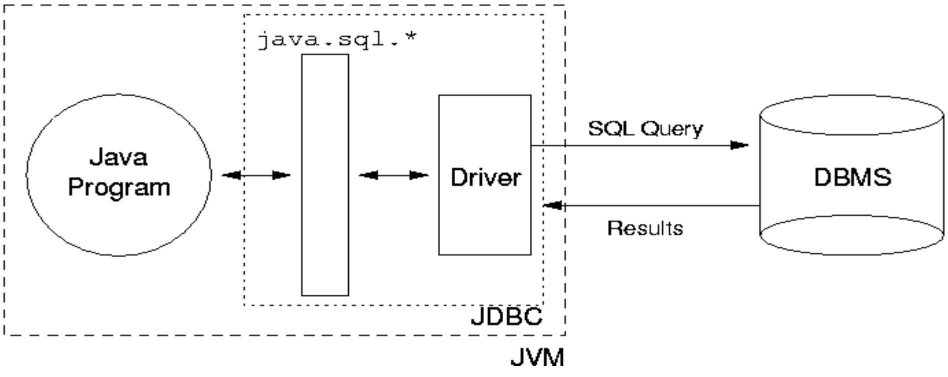
1.4 Detail Description of Technology Used:

1. **<u>JAVA:</u>**

Java is a general-purpose, object-oriented programming language developed by Sun Microsystems of USA in 1991.Originally called Oak by James Gosling (one of the inventor of the language). Java was invented for the development of software for cunsumer electronic devices like TVs, tosters, etc. The main aim had to make java simple, portable and reliable.

Java Authors: James, Arthur Van, and others. Java is a high-level, third generation programming language, like C, FORTRAN, Smalltalk, Perl, and many others. You can use Java to write computer applications that play games, store data or do any of the thousands of other things computer software can do. Compared to other programming languages, Java is most similar to C. However although Java shares much of C's syntax, it is not C. Knowing how to program in C or, better yet, C++, will certainly help you to learn Java more quickly, but you don't need to know C to learn Java. A Java compiler won't compile C code, and most large C programs need to be changed substantially before they can become Java programs. What's most special about Java in relation to other programming languages is that it lets you write special programs called applets ,web project etc. that can be downloaded from the Internet and played safely within a web browser. Java language is called as an Object-Oriented Programming language and before beginning for Java, we have to learn the concept of OOPs (Object-OrientedProgramming).

JDBC DRIVER MODEL



In the commercial world, we use Java 2 Enterprise Edition (J2EE) to solve business problems, to develop commercial software, or to provide contract services to other businesses' projects. If a company wants to build an e-business Website using a multitier architecture, it usually involves managers, architects, designers, programmers, testers, and database experts throughout the development lifecycle

Introduction to Tomcat web server

Tomcat is an open source web server developed by Apache Group. Apache Tomcat is the servlet container that is used in the official Reference Implementation for the Java Servlet and Java Server Pages technologies. The Java Servlet and Java Server Pages specifications are developed by Sun under the Java Community Process. Web Servers like Apache Tomcat support only web components while an application server supports web components as well as business components (BEAs Web logic, is one of the popular application server).To develop a web application with jsp/servlet install any web server like JRun, Tomcat etc to run your application.

Mapping

Mapping Java classes to database tables is accomplished through the configuration of an XML file or by using Java Annotations. When using an XML file, Hibernate can generate skeletal source code for the persistence classes. This is unnecessary when annotations are used. Hibernate can use the XML file or the annotations to maintain the database schema.

Facilities to arrange one-to-many and many-to-many relationships between classes are provided. In addition to managing associations between objects, Hibernate can also manage reflexive associations where an object has a one-to-many relationship with other instances of its own type.

MySQL:

MySQL is the world's most used open source relational database management system (RDBMS) as of 2008 that runs as a server providing multi-user access to a number of databases.The MySQL development project has made its source code available under the terms of the GNU General Public License, as well as under a variety of proprietary agreements. MySQL was owned and sponsored by a single for-profit firm, the Swedish company MySQL AB, now owned by Oracle Corporation. MySQL is a popular choice of database for use in web applications, and is a central component of the widely used LAMP open source web application software stack (and other 'AMP' stacks). LAMP is an acronym for "Linux, Apache, MySQL, Perl/PHP/Python." Free-software-open source projects that require a full-featured database management system often use MySQL.

Interfaces

MySQL is a relational database management system (RDBMS), and ships with no GUI tools to administer MySQL databases or manage data contained within the databases. Users may use the included command line tools, or use MySQL "front-ends", desktop software and web applications that create and manage MySQL databases, build database structures, back up data, inspect status, and work with data records. The official set of MySQL front-end tools, MySQL Workbench is actively developed by Oracle, and is freely available for use.

Graphical

The official MySql Workbench is a free integrated environment developed by MySQL AB, that enables users to graphically administer MySQL databases and visually design database structures. MySQL Workbench replaces the previous package of software, MySQL GUI Tools. Similar to other third-party packages, but still considered the authoritative MySQL front end, MySQL Workbench lets users manage database design & modeling, SQL development (replacing MySQL Query Browser) and Database administration (replacing MySQL Administrator).

MySQL Workbench is available in two editions, the regular free and open source *Community Edition* which may be downloaded from the MySQL website, and the proprietary *Standard Edition* which extends and improves the feature set of the Community Edition.

2 : PROPOSED SYSTEM

2.1 Proposed System

The Proposed system ensures the complete freedom for users, where user by themselves can login easily and can book his/her ticket. In the proposed system all the parameter are considered to maintain neat and easier solution. Now our proposed system will overcome all the drawbacks of existing system. The proposed system due to computerized is much faster in reservation process, cancellation process and transactions. Our proposed system allows only registered users to book the tickets, view the booked tickets and cancel their ticket . In this Proposed system ticket generation is also done for a booked ticket as well as cancelled ticket. Even admin can login and admin can maintain /modify the details of flights in our proposed system.

2.2 Objectives of System

- It is very user-friendly and having added more features.
- Booking tickets and cancellations of tickets from any where to any place.
- Data Security and Ensure data accuracy's.
- Faster technique to generate flight ticket.
- Provide convenience to travellers.
- Better service and Minimum time require

2.3 User Requirements

2.3.1: Functional Requirements

1) software engineering, a functional requirement defines a function of a software system or its component.

2) A function is described as a set of inputs, the behavior, and outputs.

3) Functional requirements may be calculations, technical details, data manipulation and processing and other specific functionality that define what a system is supposed to accomplish. Behavioural requirements describing all the cases where the system uses the functional requirements are captured in use cases.

4) Functional requirements are supported by non-functional requirements (also known as quality requirements), which impose constraints on the design or implementation (such as performance requirements, security, or reliability). Generally, functional requirements are expressed in the form "system must do <requirement>", while non-functional requirements are "system shall be <requirement>". The plan for implementing functional requirements is detailed in the system design. The plan for implementing non-functional requirements is detailed in the system architecture.

5) As defined in requirements engineering, functional requirements specify particular results of a system. This should be contrasted with non-functional requirements which specify overall characteristics such as cost and reliability. Functional requirements drive the application architecture of a system, while non-functional requirements drive the technical architecture of a system

2.3.2: Non-Functional Requirements

Product Requirements

A) Usability requirements

Usability is the ease of use and learns ability of a human-made object. The object of use can be a software application, website, book, tool, machine, process, or anything a human interacts with. A usability study may be conducted as a primary job function by a usability analyst or as a secondary job function by designers, technical writers, marketing personnel, and others. Usability includes methods of measuring usability, such as needs analysis and the study of the principles behind an object's perceived efficiency or elegance. In human computer interaction and computer science, usability studies the elegance and clarity with which the interaction with a computer program or a web site (web usability) is designed. Usability differs from user satisfaction and user experience because usability also considers Usefulness.

B) Reliability requirements

Reliability deals with the study, evaluation, and lifecycle management of reliability: the ability of a system or component to perform its required functions under stated conditions for a specified period of time. Reliability engineering is a sub-discipline within systems engineering. Reliability is theoretically defined as the probability of failure, the frequency of failures, or in terms of availability, a probability derived from reliability and maintainability. Maintainability and maintenance may be defined as a part of reliability engineering. Reliability plays a key role in cost-effectiveness of systems.

C) Portability requirements

Portability in high-level computer programming is the usability of the same software in different environments. The pre-requirement for portability is the generalized abstraction between the application logic and system interfaces. When software with the same functionality is produced for several computing platforms, portability is the key issue for development cost reduction.

D) Efficiency requirements

Resource consumption for given load describes efficiency of product and web site.

E) Performance requirements

Performance metrics include availability, response time, channel capacity, latency, completion time, service time, bandwidth, throughput, relative efficiency, scalability, performance per watt, compression ratio, instruction path length and speed up.

Organizational Requirements

A) Implementation requirements

Implementation is the realization of an application, or execution of a plan, idea, model, design, specification, standard, algorithm, or policy.

An implementation is a realization of a technical specification or algorithm as a program, software component, or other computer system through programming and deployment. Many implementations may exist for a given specification or standard. For example, web browsers contain implementations of World Wide Web Consortium – recommended specifications, and software development tools contain implementations of programming languages.

B) Standard requirements

The project should be developed as per standard format specified by IEEE.

Typical platforms include a computer architecture, operating system, programming languages and related user interface. The product should be developed as per client's standard requirements.

External Requirements

A) Interoperability requirements

Interoperability is a property of a product or system, whose interfaces are completely understood, to work with other products or systems, present or future,
without any restricted access or implementation.

The IEEE Glossary defines interoperability as:
the ability of two or more systems or components to exchange information and to use the information that has been exchanged.

B) Legislative requirements

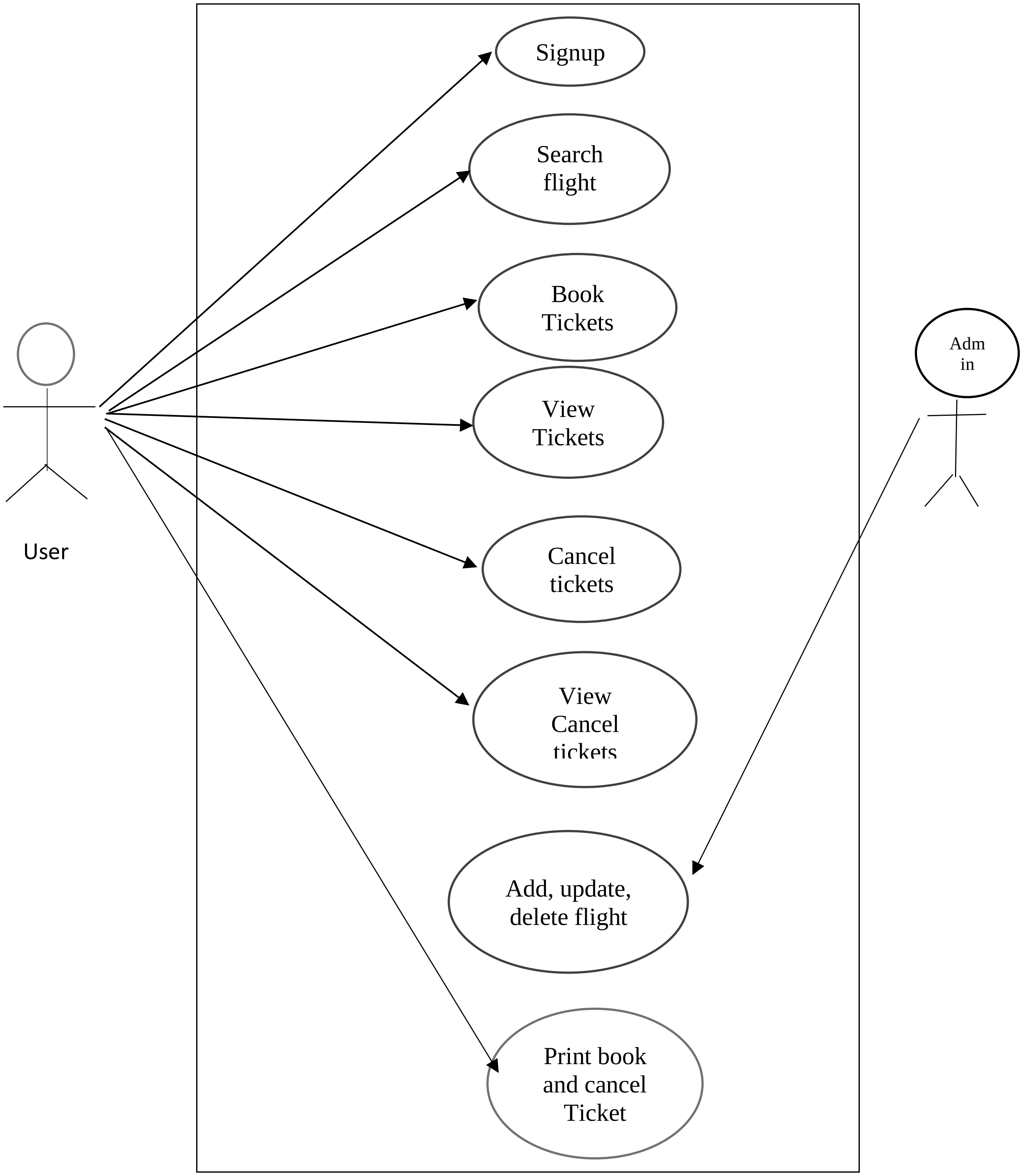
In the proprietary software industry, an end-user license agreement or software license agreement is the contract between the licensor and purchaser, establishing the purchaser's right to use the software. The license may define ways under which the copy can be used. Software companies often make special agreements with large businesses and government entities that include support contracts and specially drafted warranties.

C) Privacy requirements

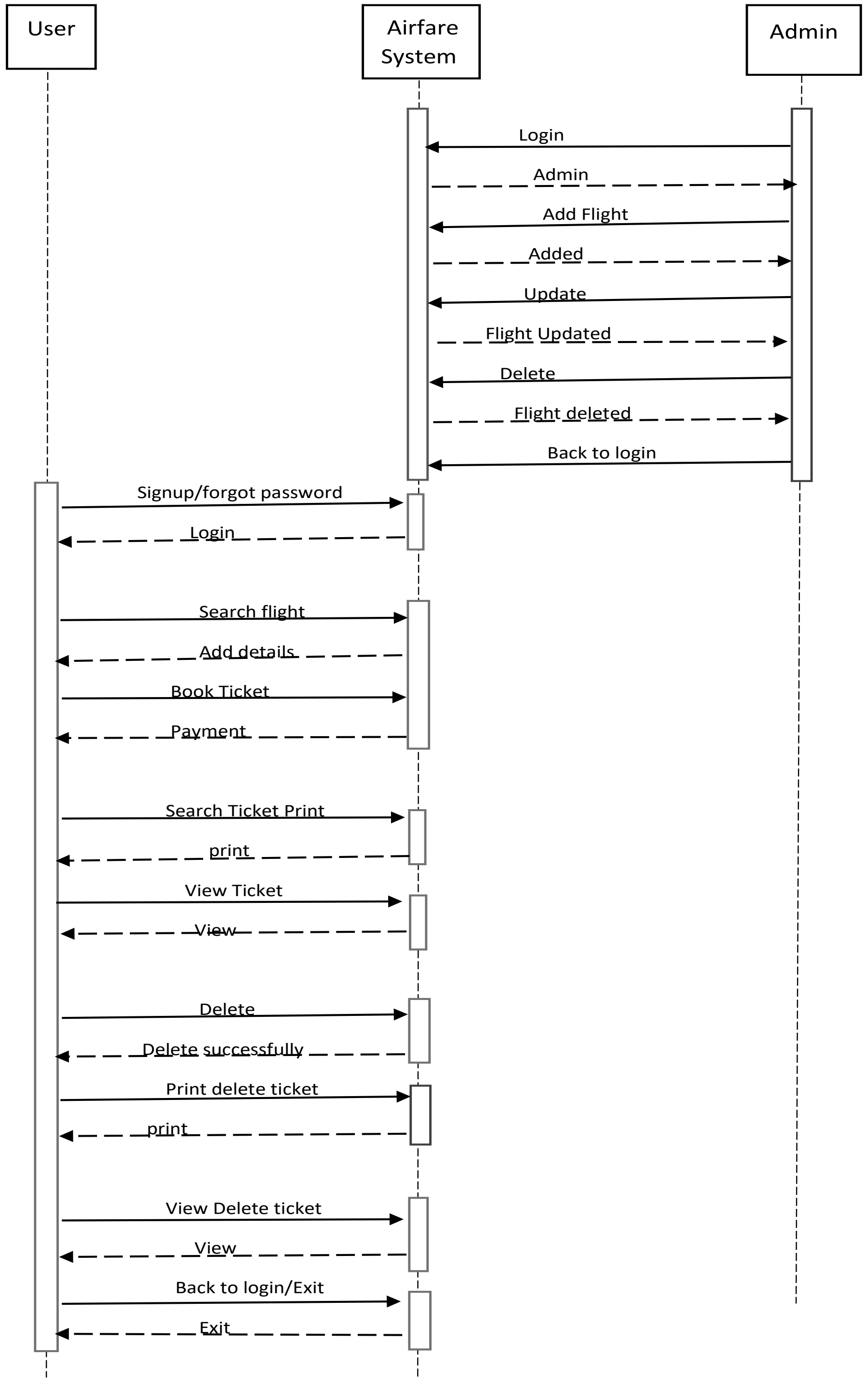
The term "privacy" means many things in different contexts. Different people, cultures, and nations have a wide variety of expectations about how much privacy a person is entitled to or what constitutes an invasion of privacy. Privacy is the ability of an individual or group to seclude themselves or information about themselves and thereby reveal themselves selectively. The boundaries and content of what is considered private differ among cultures and individuals, but share basic common themes. Privacy is sometimes related to anonymity, the wish to remain unnoticed or unidentified in the public realm.

3 : ANALYSIS & DESIGN

3.1 Use case diagram

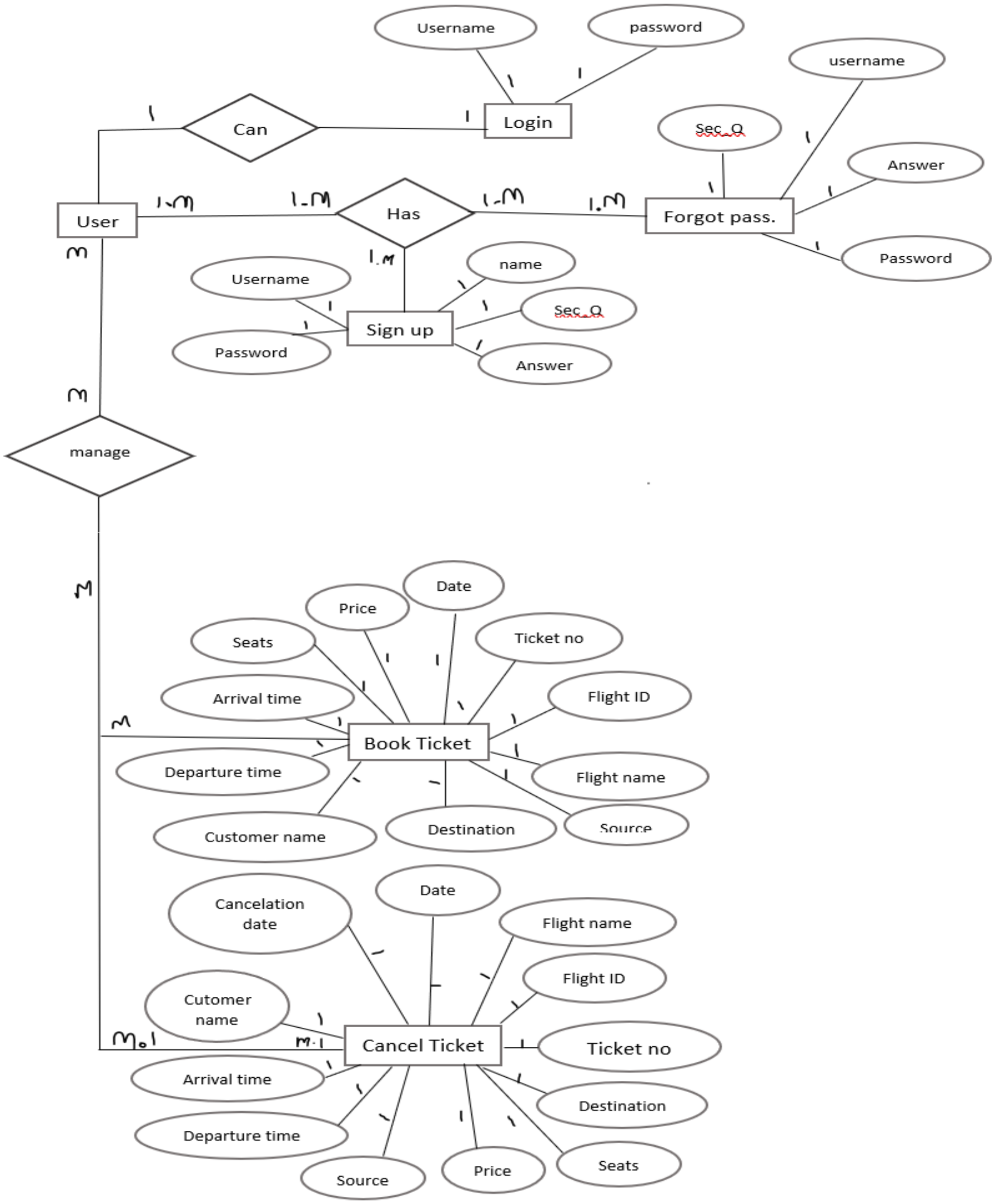


3.2 Sequence Diagram

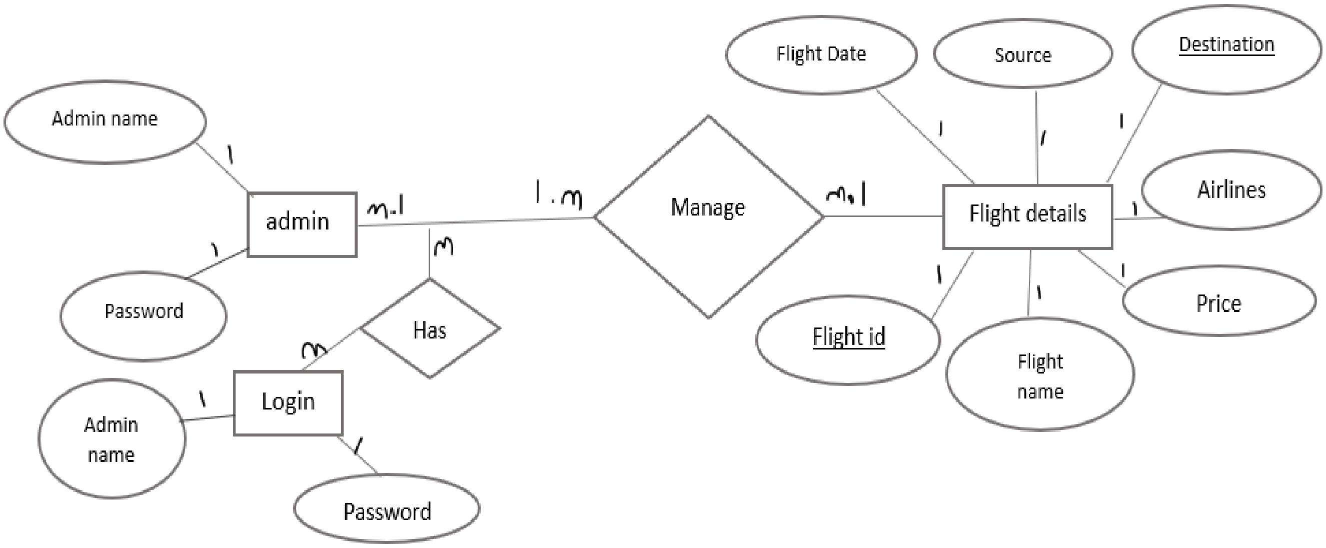


3.3 ER Diagram

For User



For Admin


Admin name
admin
Password
Login
Admin name
Password
Has
Manage
Flight details
Flight Date
Source
Destination
Airlines
Price
Flight id
Flight name
m.1
1.m
m
m
m.1
1

3.4 DF diagram

DFD 0 level :

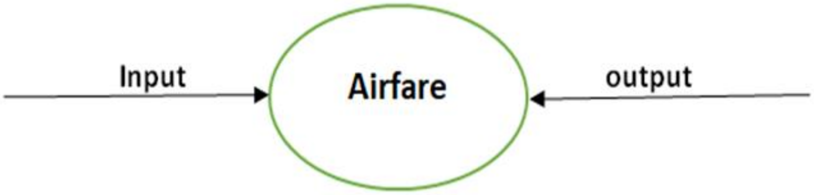
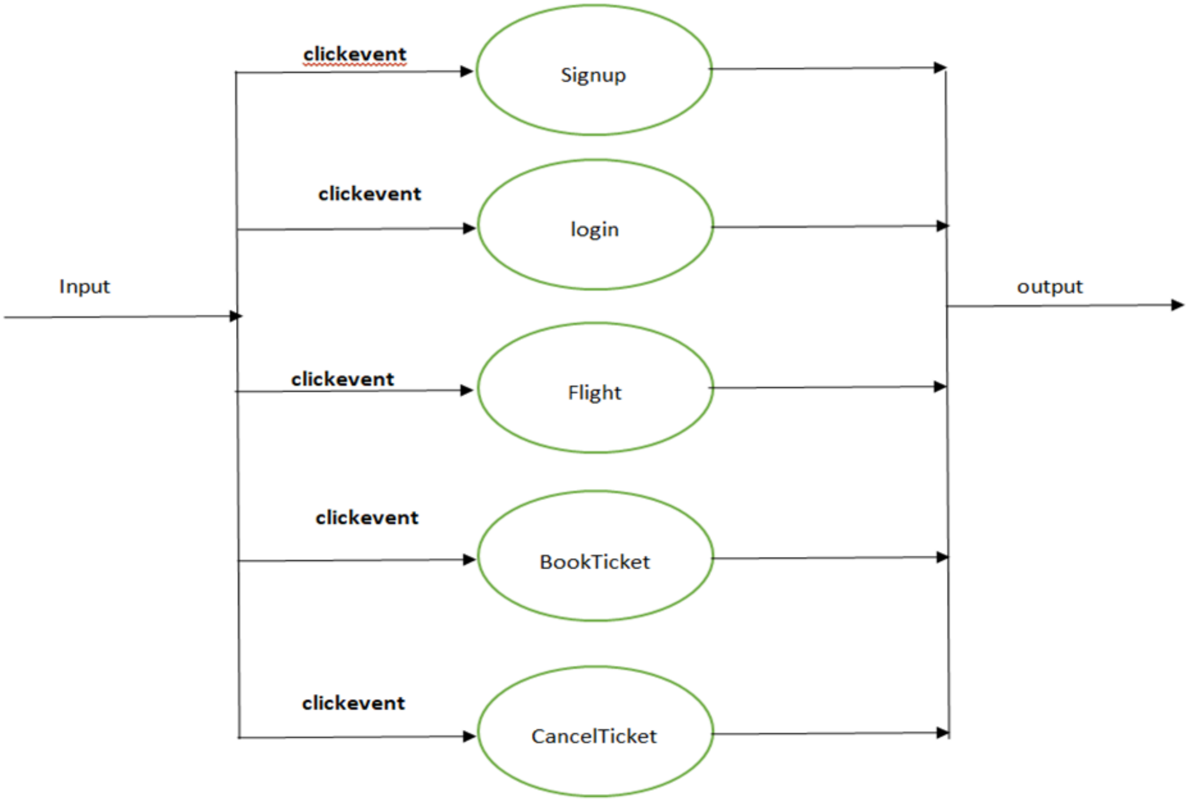


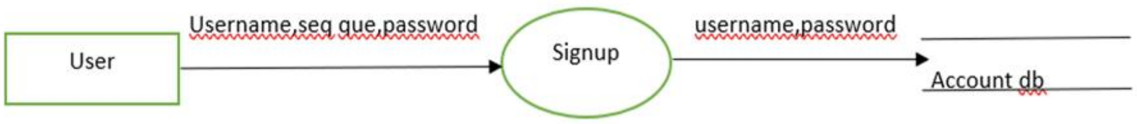
Fig. DFD 0 level.

DFD 1st Level:

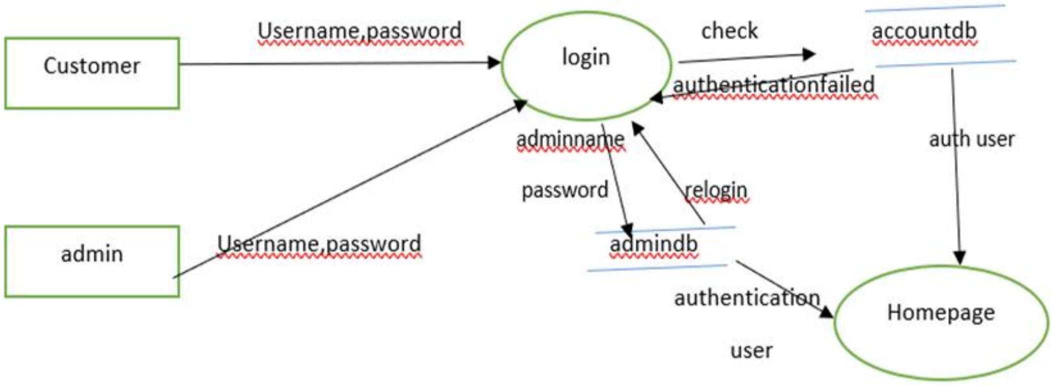


DFD 2nd Level

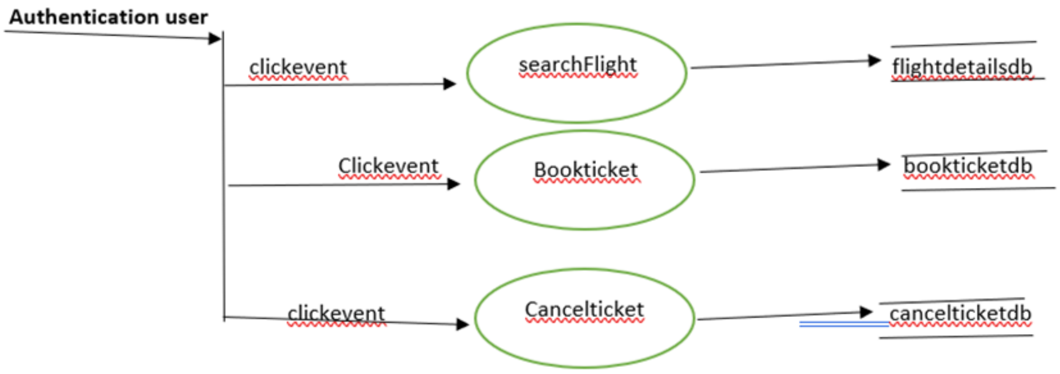
Level (a)



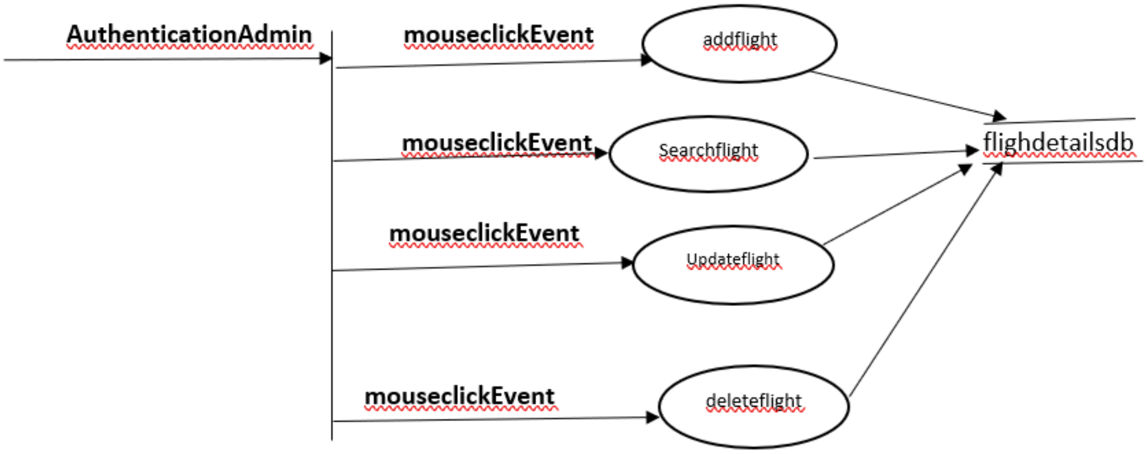
Level 2 (b)



Level 2 (c)

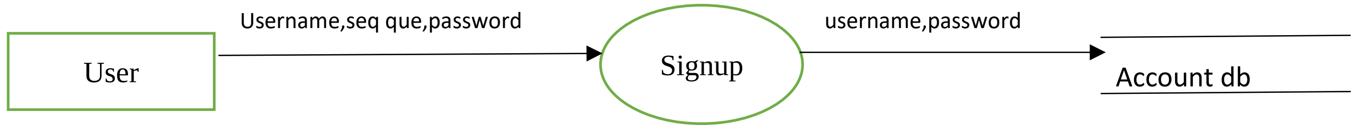


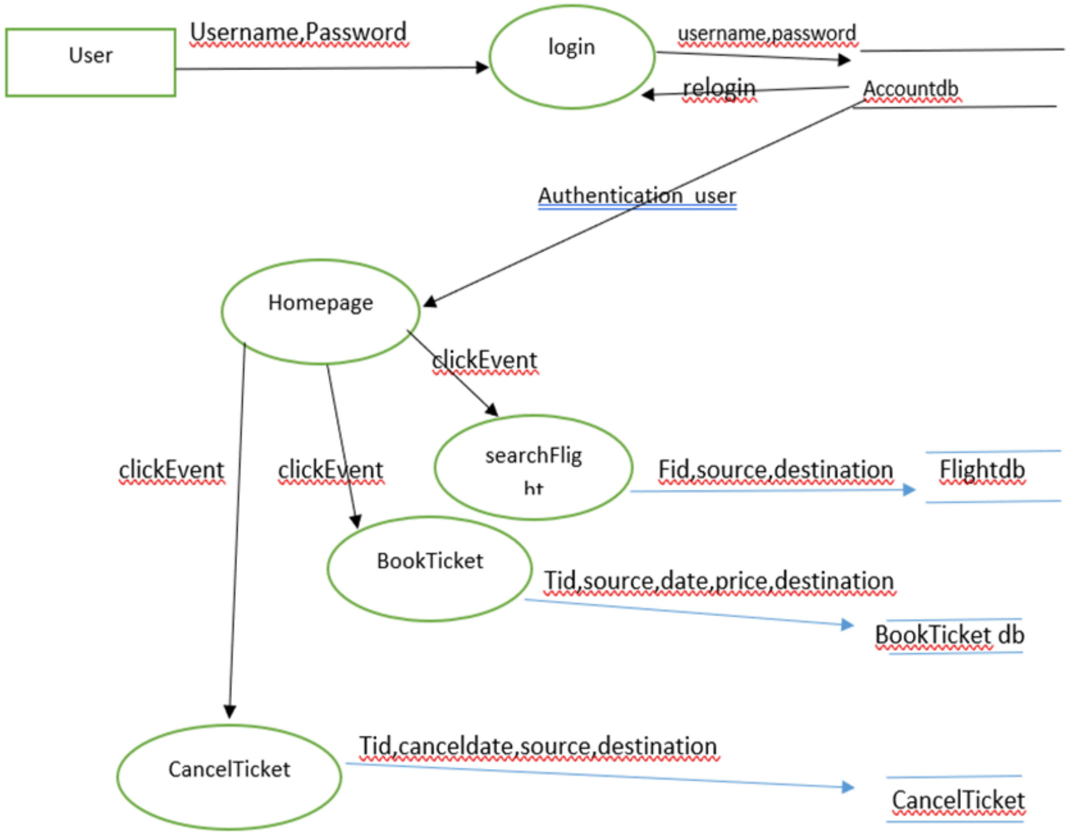
Level 2(d)

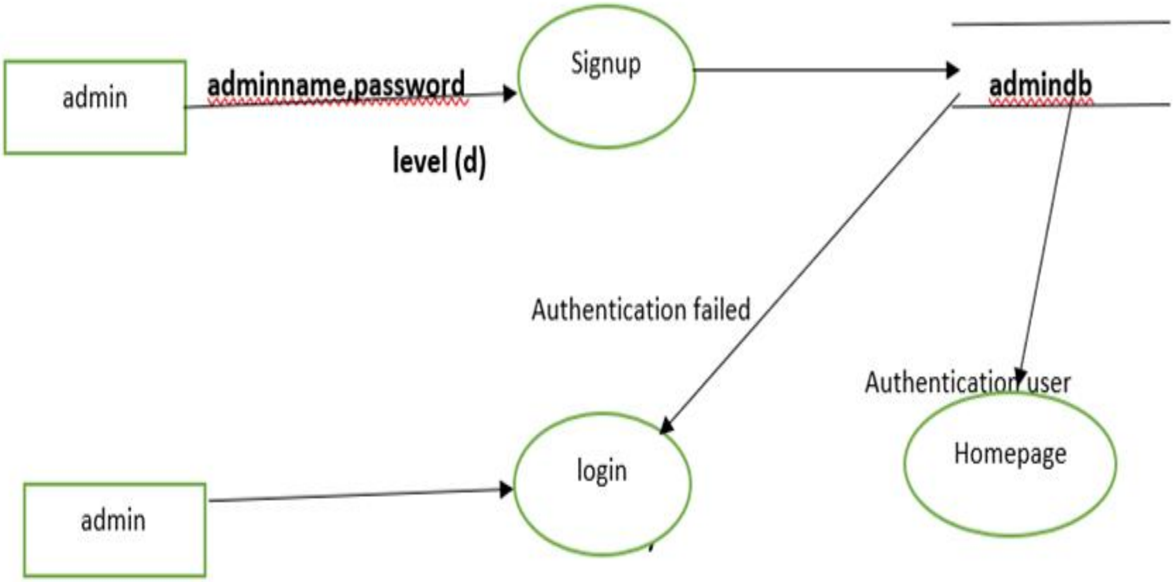


DFD 2nd level diagram

DFD 3rd level







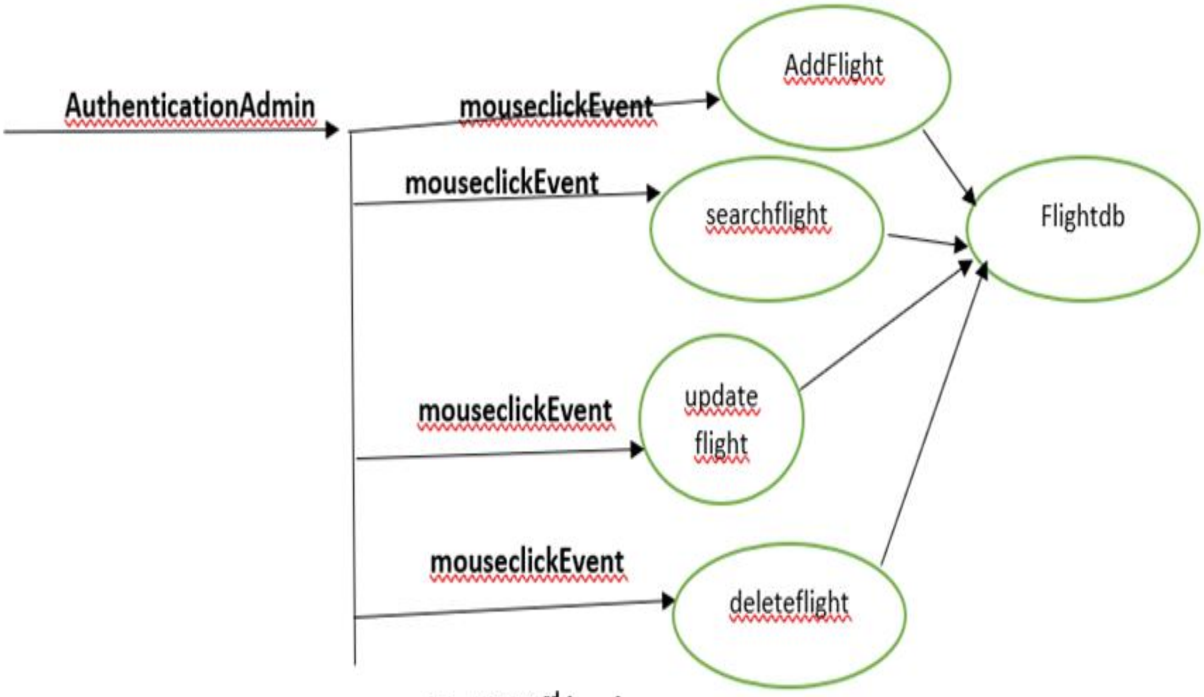
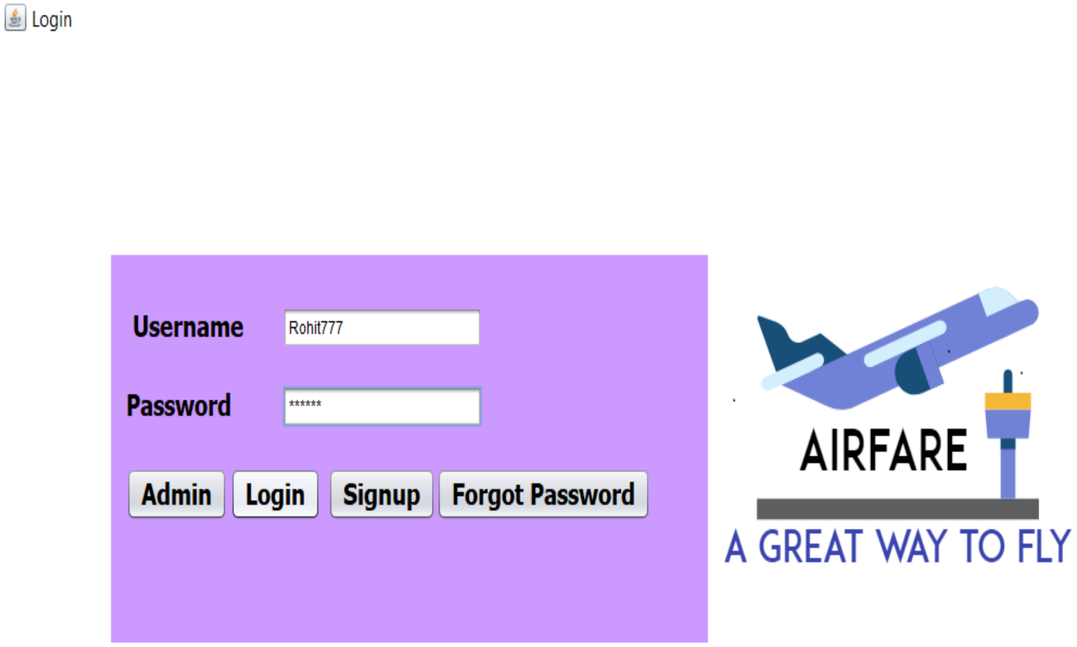


Fig. DFD 3rd level

3.5 User Interface Design (Screens etc.)

Login Page:

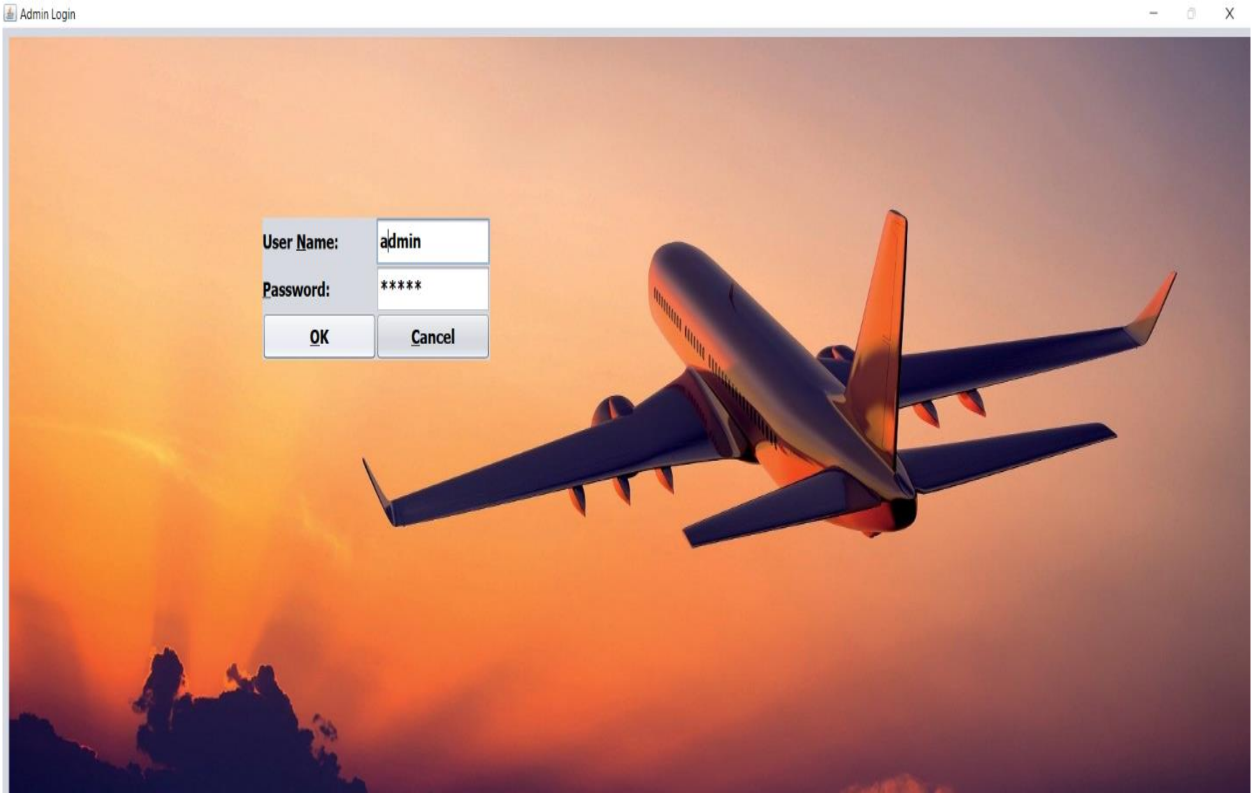


Login

Username: Rohit777

Password: ******

Admin Login :

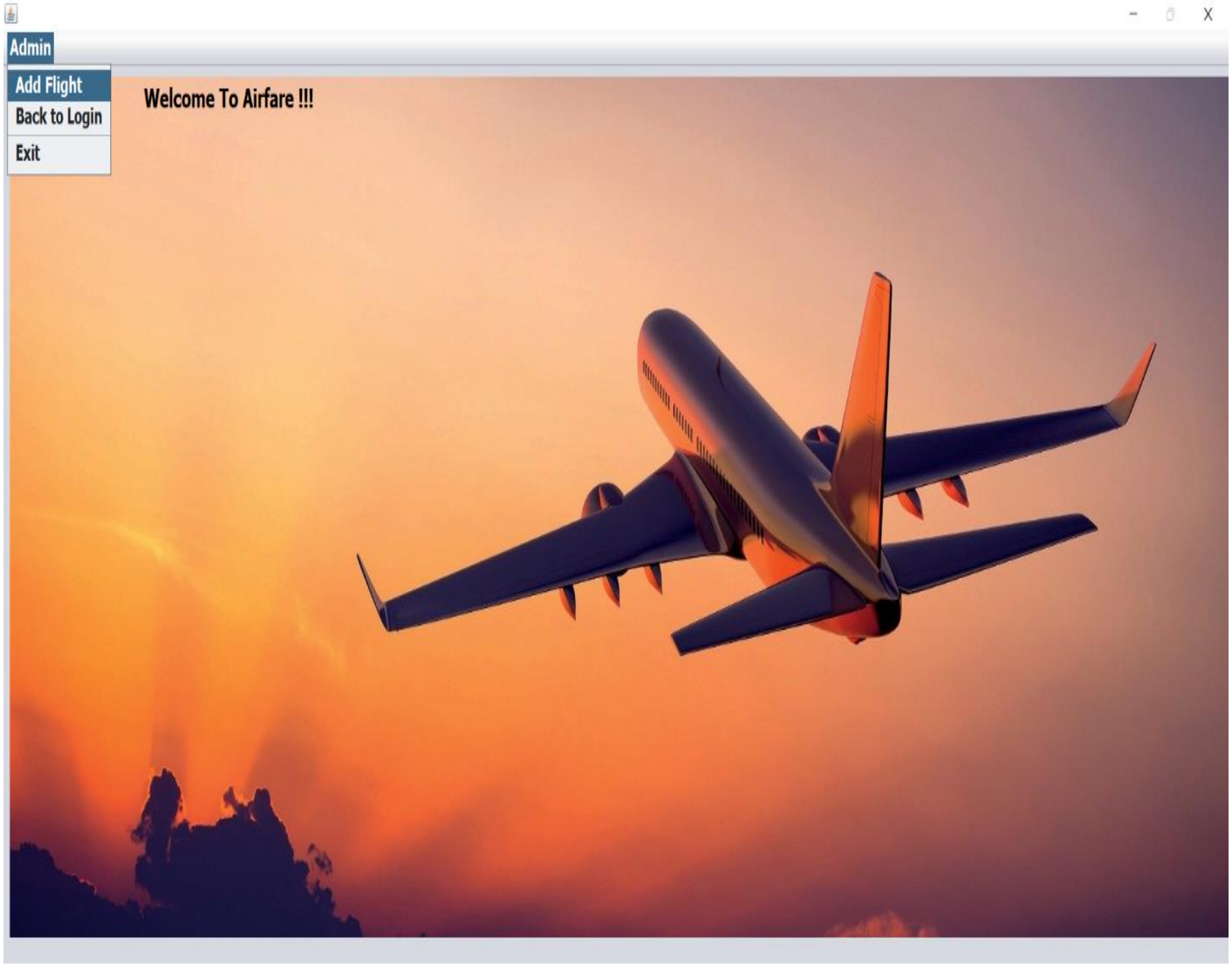


Admin Login

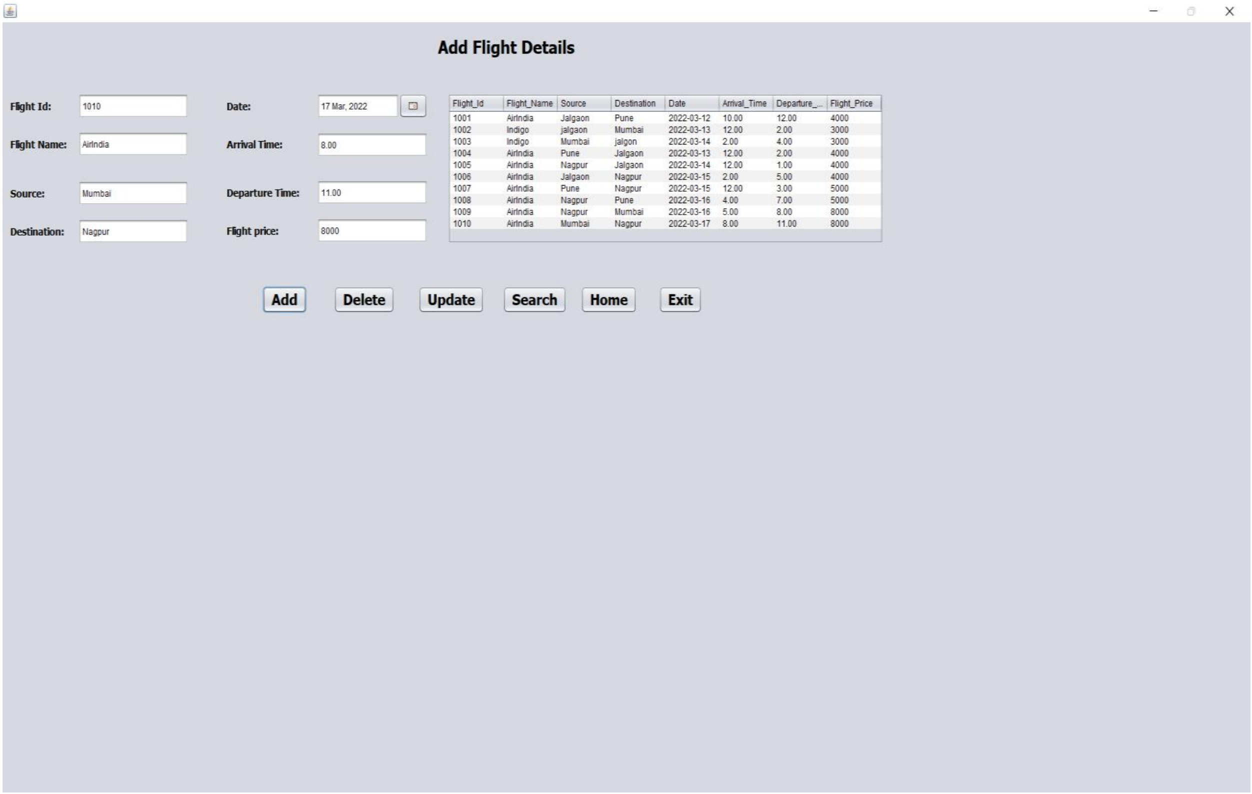
User Name: admin

Password: *****

Admin Home Page:



Add Flight :

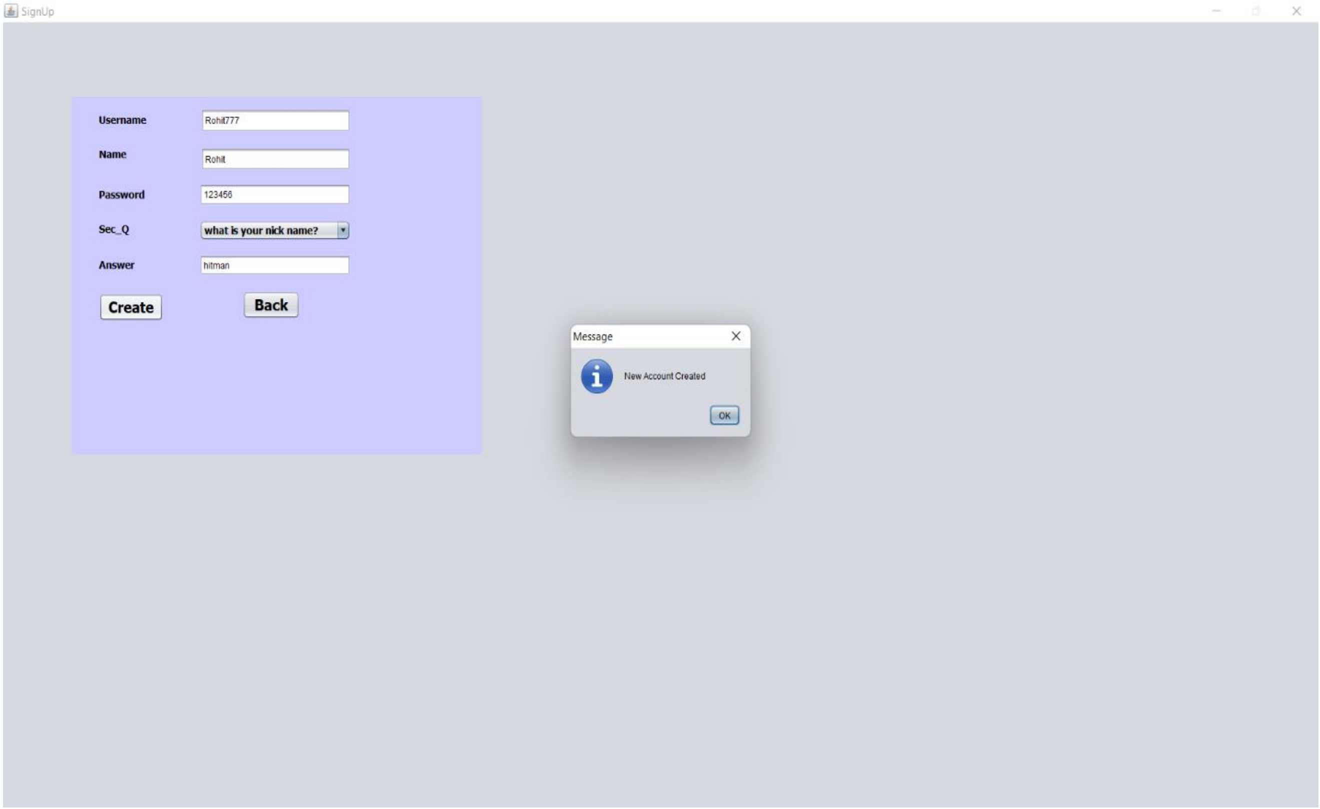


Add Flight Details

Flight Id:	1010	Date:	17 Mar, 2022
Flight Name:	AirIndia	Arrival Time:	8.00
Source:	Mumbai	Departure Time:	11.00
Destination:	Nagpur	Flight price:	8000

Flight_Id	Flight_Name	Source	Destination	Date	Arrival_Time	Departure_...	Flight_Price
1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	4000
1002	Indigo	jalgaon	Mumbai	2022-03-13	12.00	2.00	3000
1003	Indigo	Mumbai	jalgon	2022-03-14	2.00	4.00	3000
1004	AirIndia	Pune	Jalgaon	2022-03-13	12.00	2.00	4000
1005	AirIndia	Nagpur	Jalgaon	2022-03-14	12.00	1.00	4000
1006	AirIndia	Jalgaon	Nagpur	2022-03-15	2.00	5.00	4000
1007	AirIndia	Pune	Nagpur	2022-03-15	12.00	3.00	5000
1008	AirIndia	Nagpur	Pune	2022-03-16	4.00	7.00	5000
1009	AirIndia	Nagpur	Mumbai	2022-03-16	5.00	8.00	8000
1010	AirIndia	Mumbai	Nagpur	2022-03-17	8.00	11.00	8000

User Signup:



Username Rohit777

Name Rohit

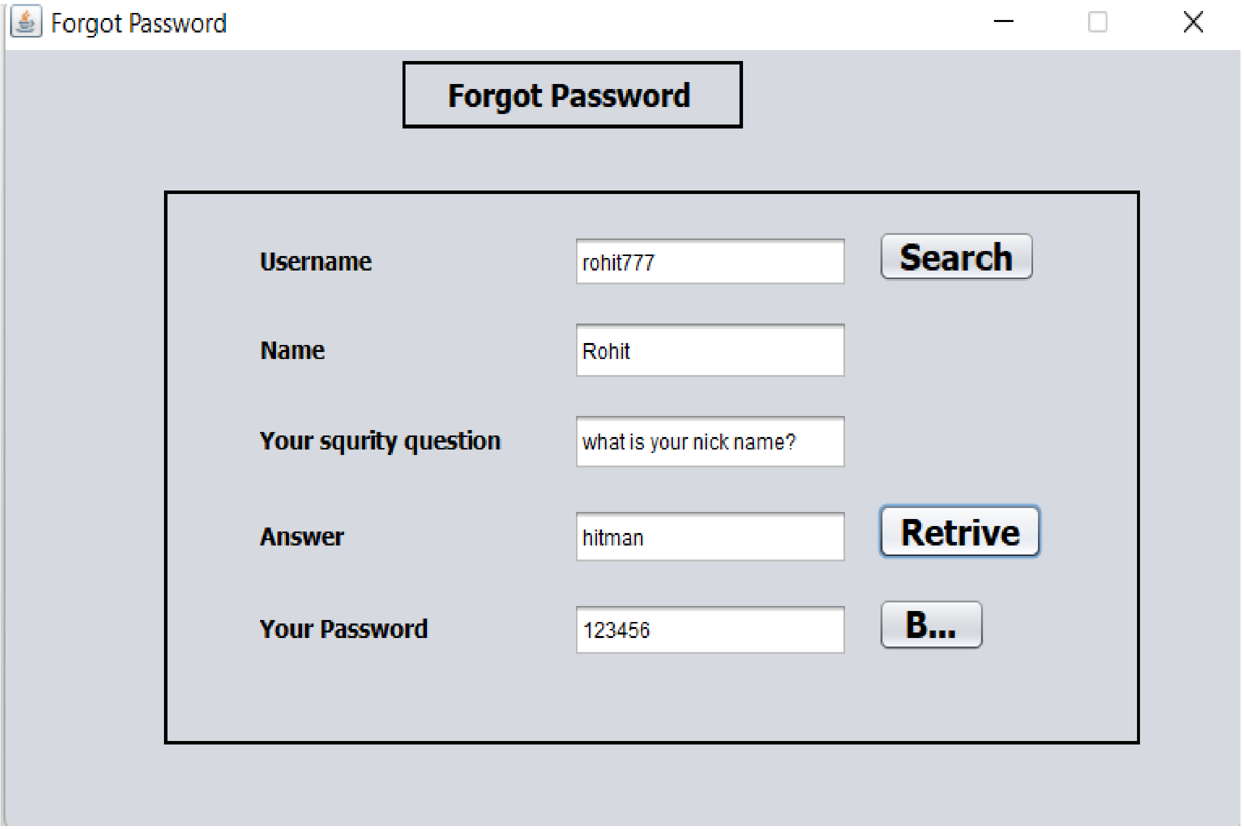
Password 123456

Sec_Q what is your nick name?

Answer hitman

New Account Created

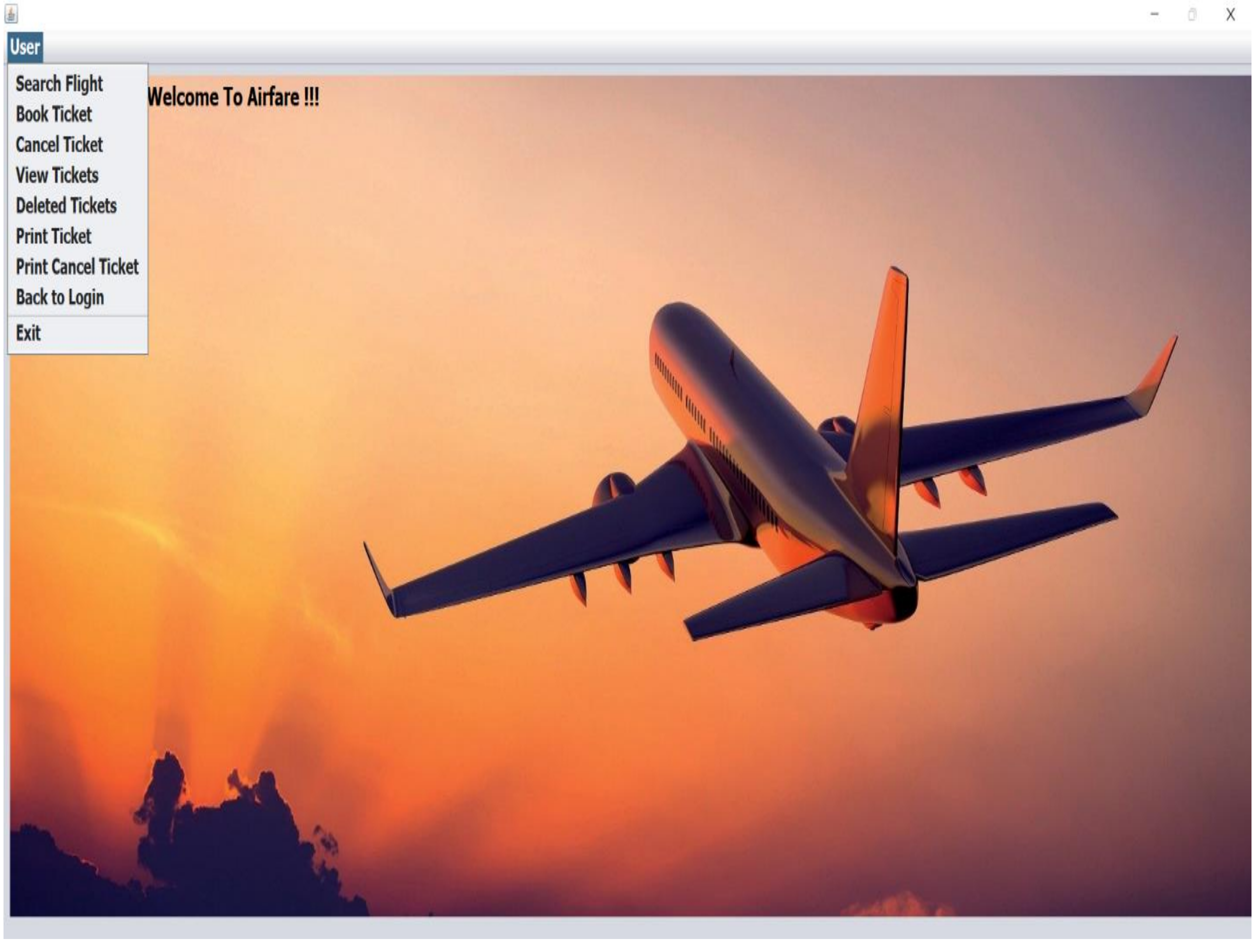
User Forgot Password:



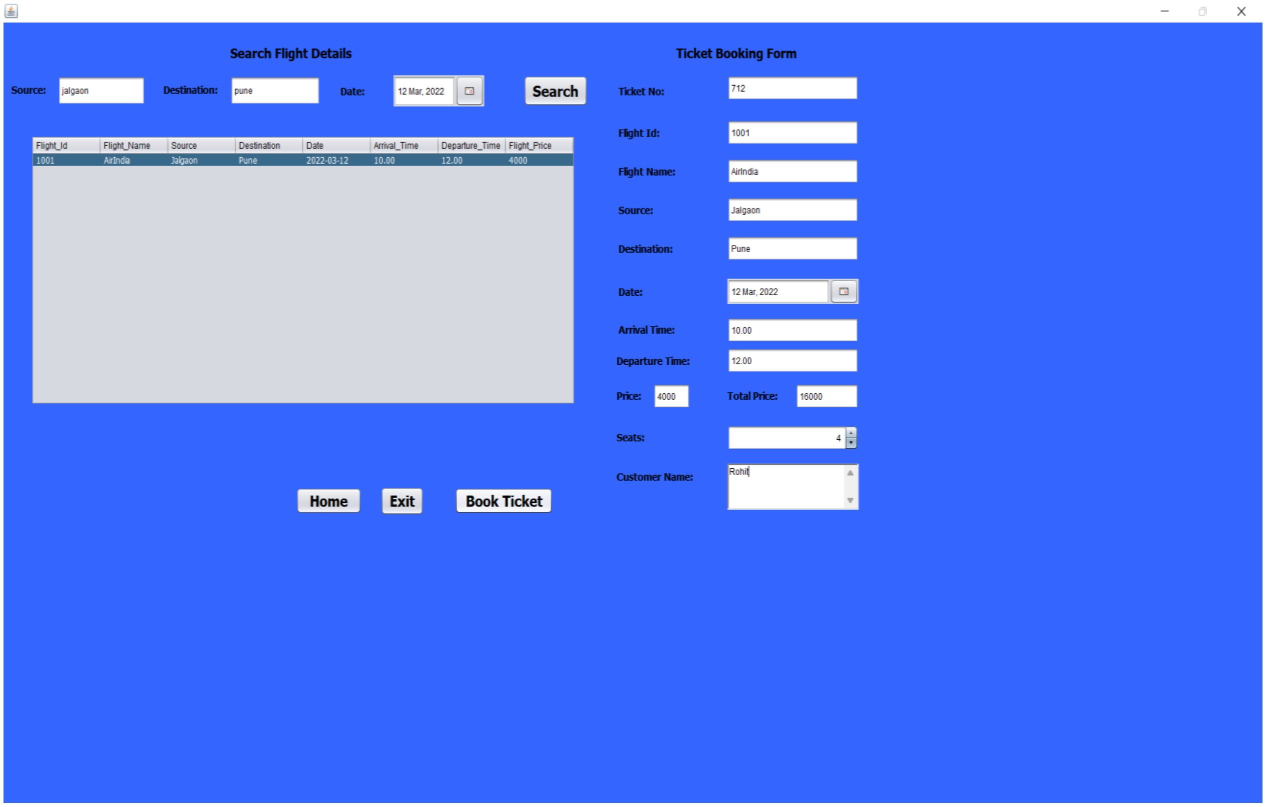
Forgot Password

Username	rohit777	Search
Name	Rohit	
Your squrity question	what is your nick name?	
Answer	hitman	Retrive
Your Password	123456	B...

User Home Page:



User search flight and book ticket :



Search Flight Details

Source: jalgaon Destination: pune Date: 12 Mar, 2022

Flight_Id	Flight_Name	Source	Destination	Date	Arrival_Time	Departure_Time	Flight_Price
1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	4000

Ticket Booking Form

Ticket No: 712

Flight Id: 1001

Flight Name: AirIndia

Source: Jalgaon

Destination: Pune

Date: 12 Mar, 2022

Arrival Time: 10.00

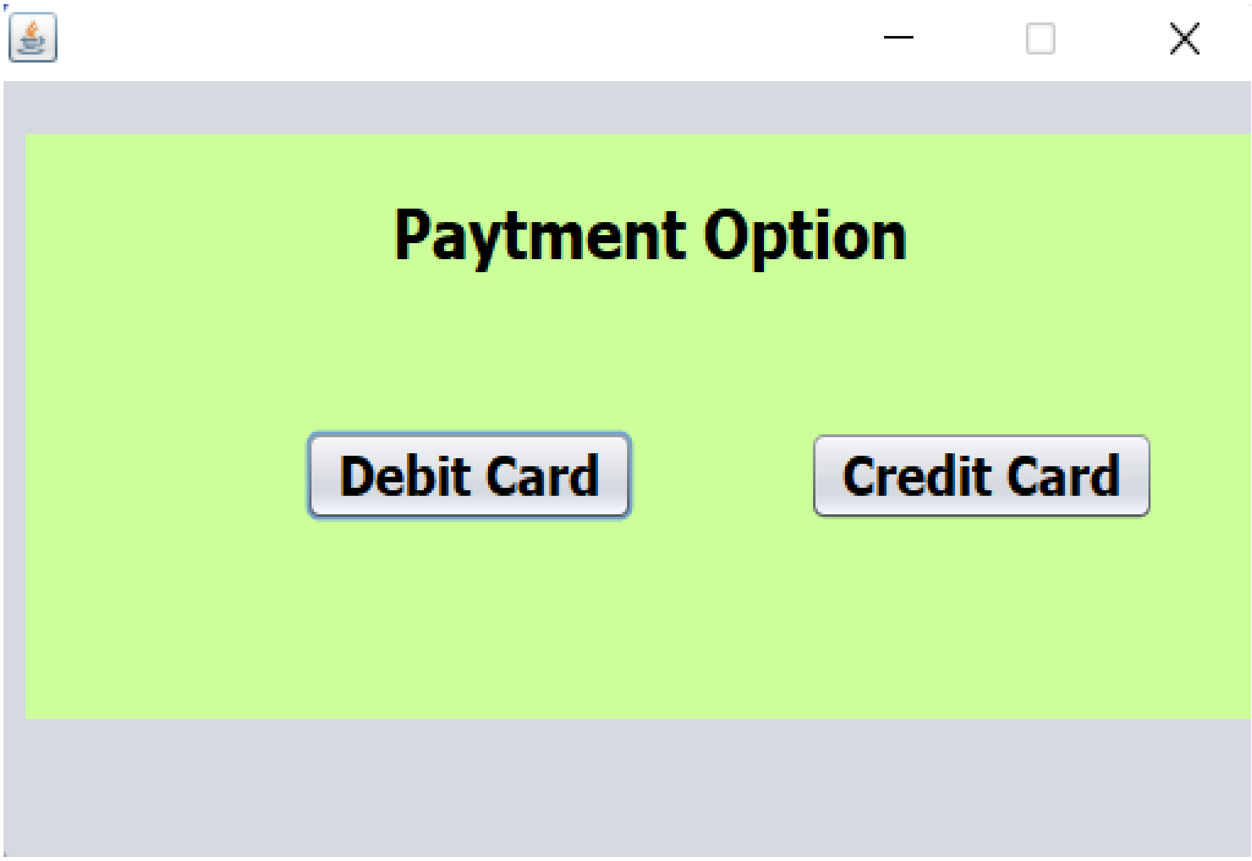
Departure Time: 12.00

Price: 4000 Total Price: 16000

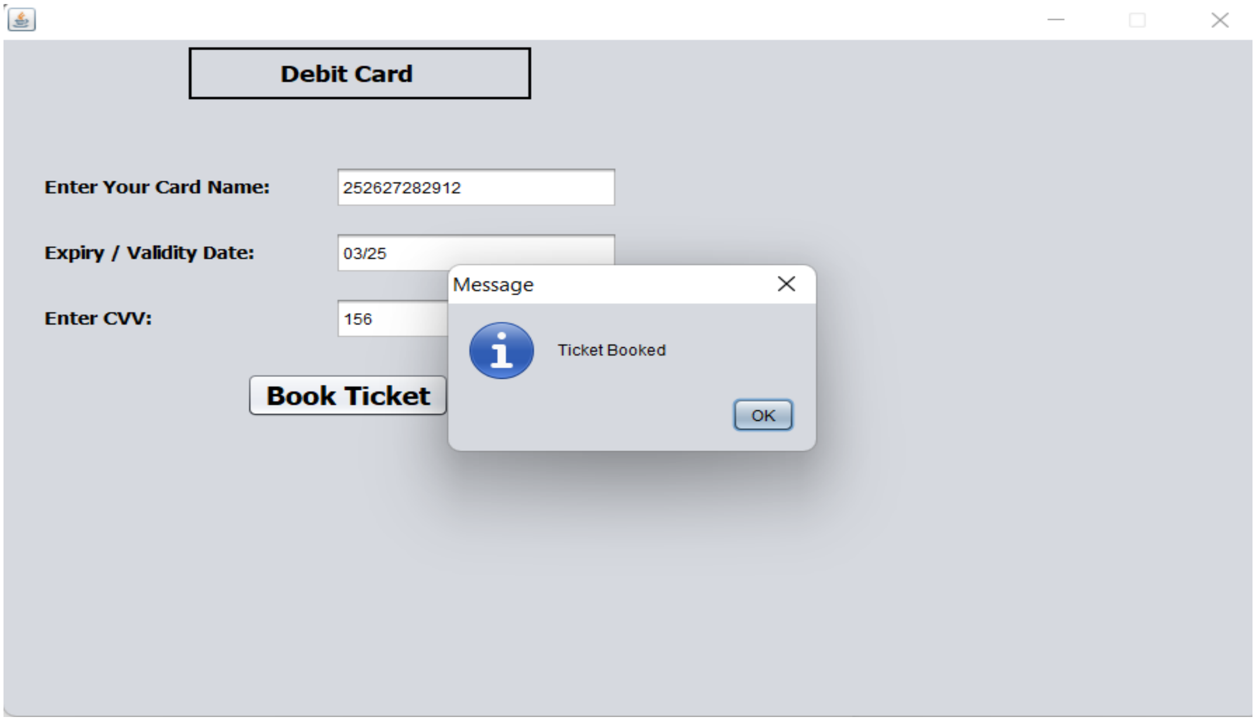
Seats: 4

Customer Name: Rohit

User Payment Option :



Debit Card:



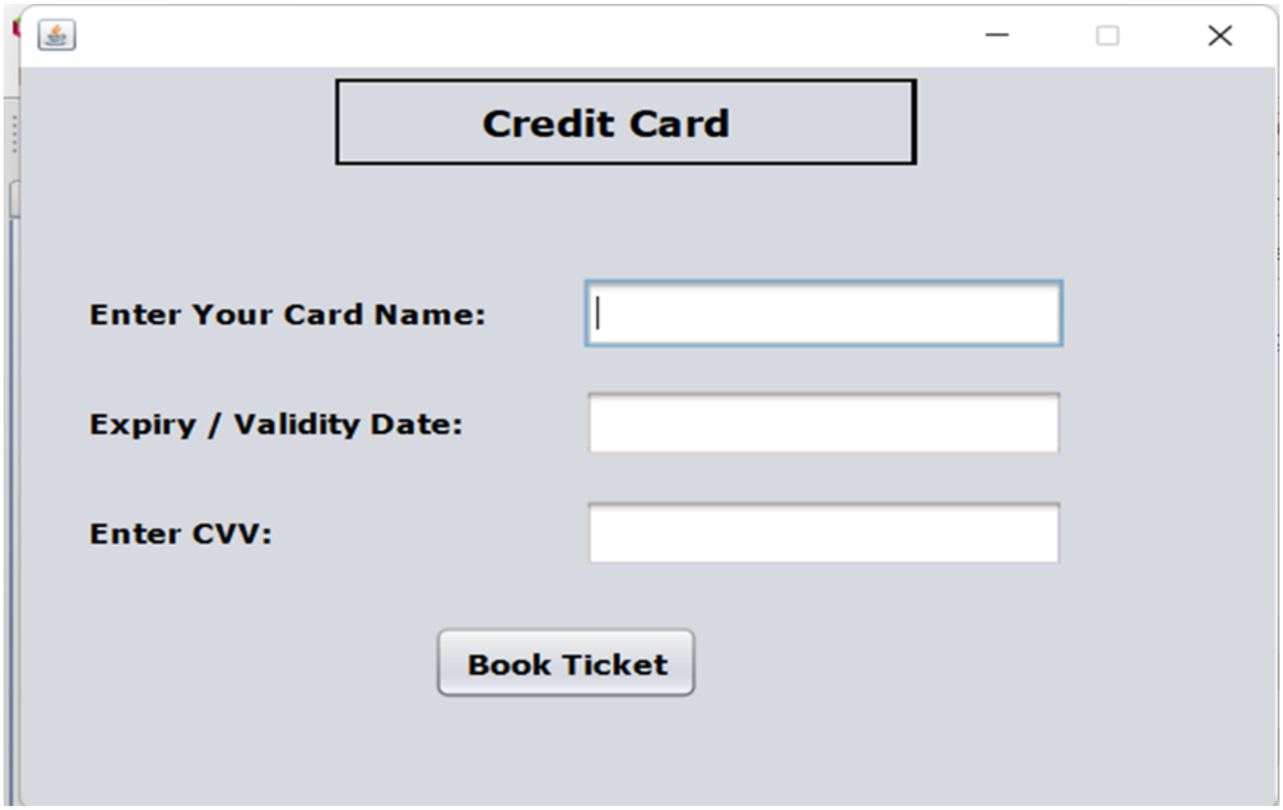
Enter Your Card Name: 252627282912

Expiry / Validity Date: 03/25

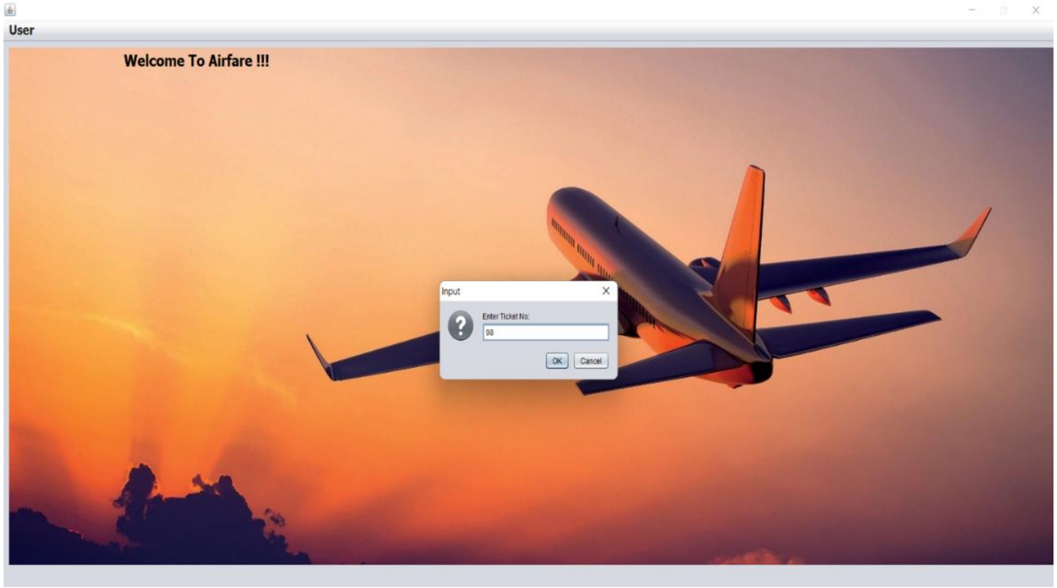
Enter CVV: 156

Ticket Booked

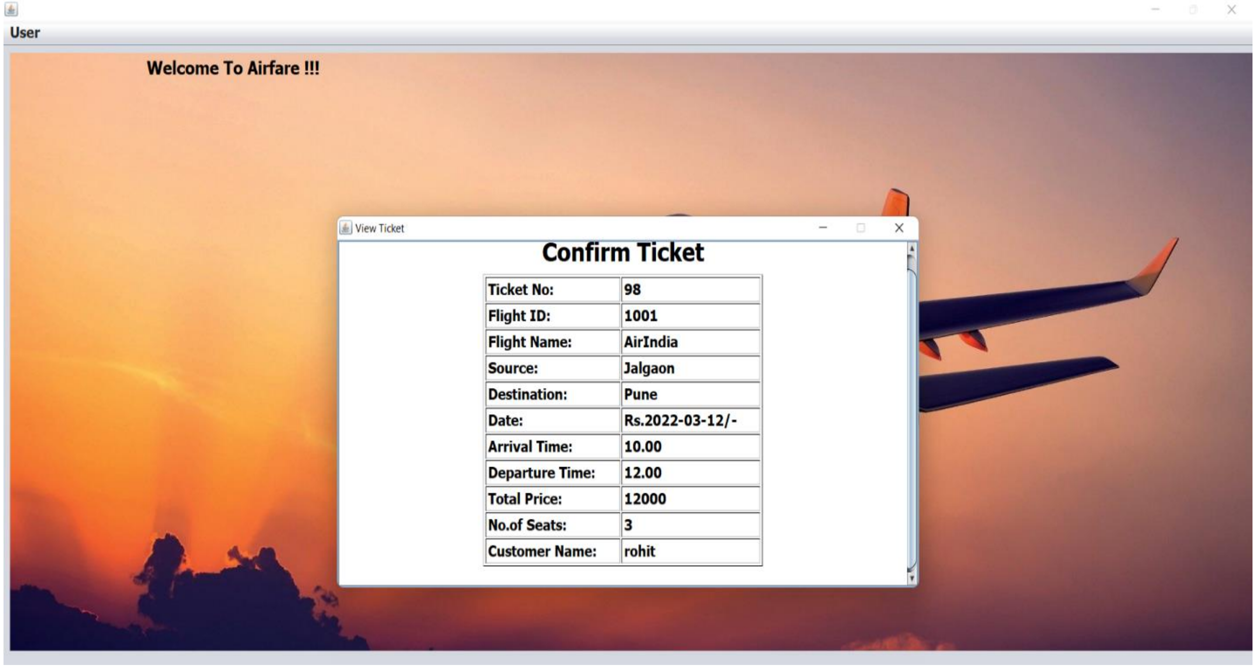
Credit Card:



Print Booked Ticket:



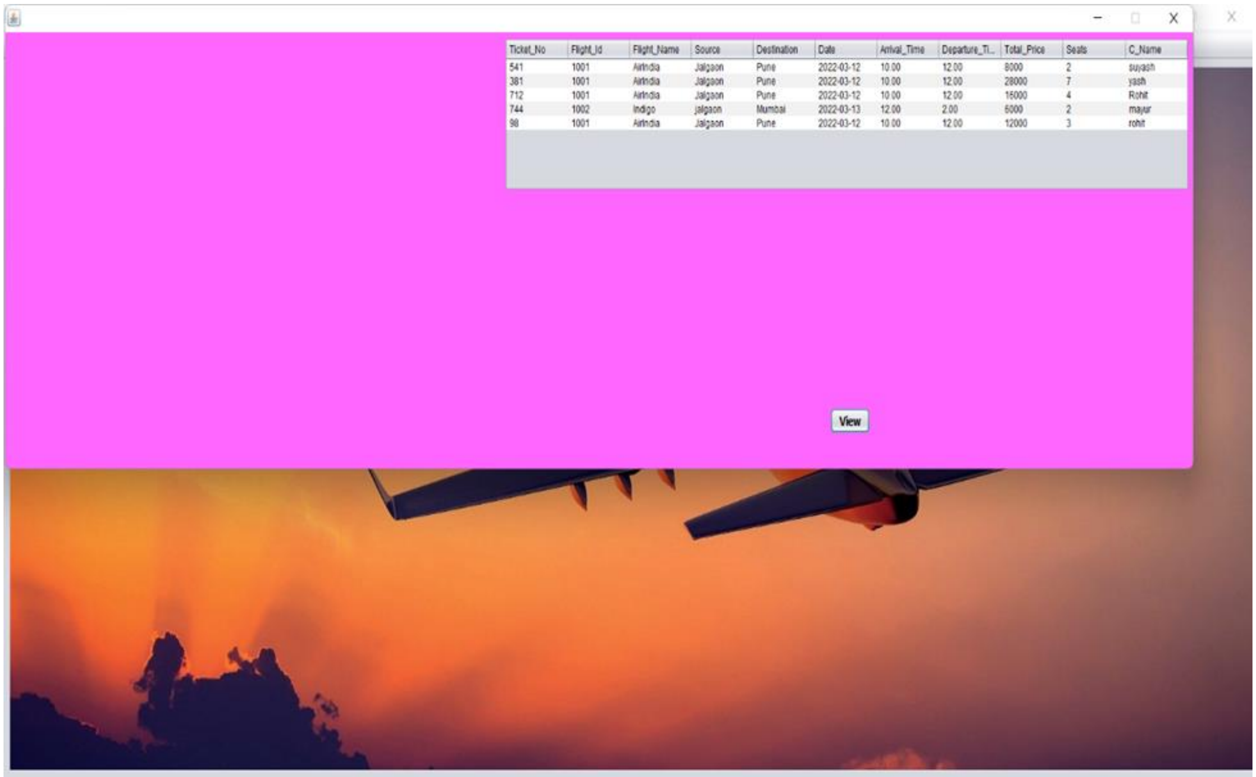
Enter Ticket No: 98



Confirm Ticket

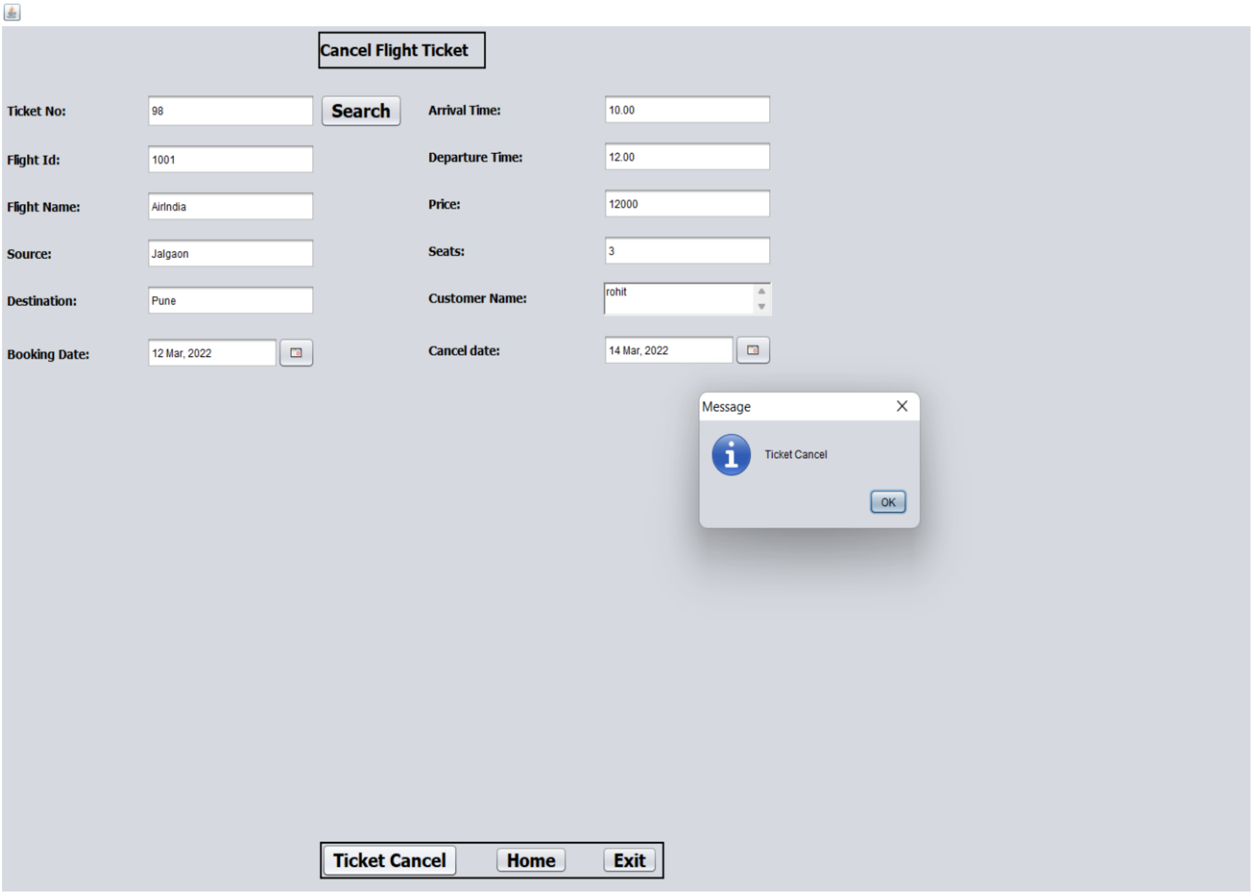
Ticket No:	98
Flight ID:	1001
Flight Name:	AirIndia
Source:	Jalgaon
Destination:	Pune
Date:	Rs.2022-03-12/-
Arrival Time:	10.00
Departure Time:	12.00
Total Price:	12000
No.of Seats:	3
Customer Name:	rohit

View Ticket:



Ticket_No	Flight_Id	Flight_Name	Source	Destination	Date	Arrival_Time	Departure_Ti...	Total_Price	Seats	C_Name
541	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	8000	2	suyash
381	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	28000	7	yash
712	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	16000	4	Rohit
744	1002	Indigo	jalgaon	Mumbai	2022-03-13	12.00	2.00	6000	2	mayur
98	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	12000	3	rohit

Ticket Cancel



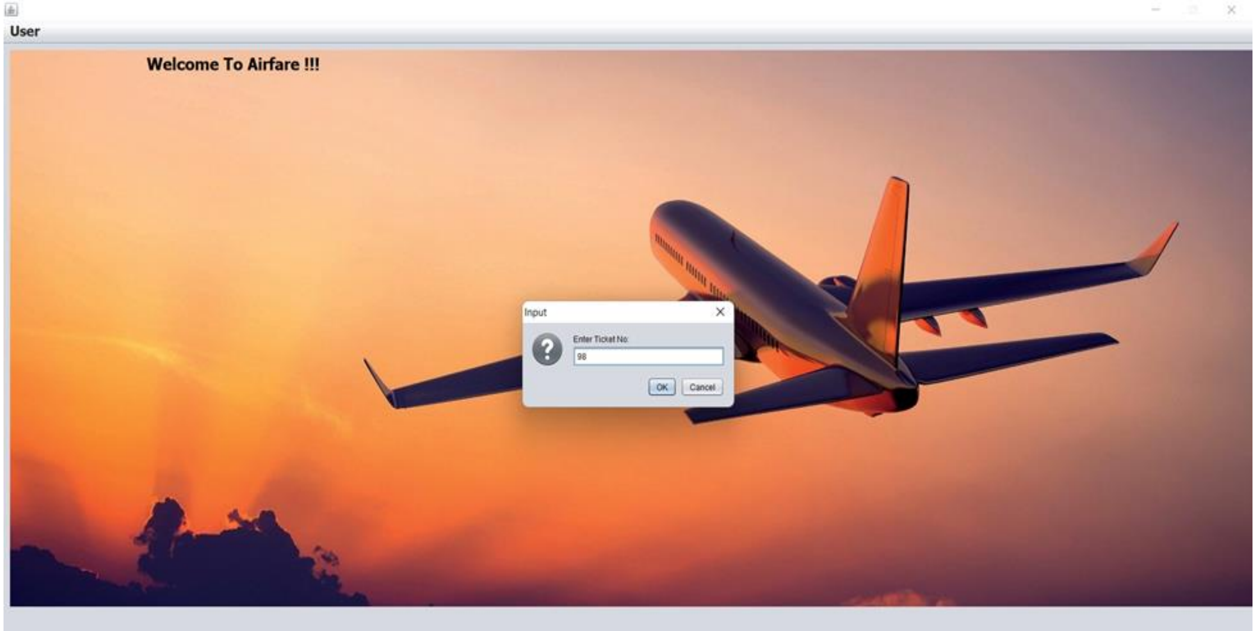
Cancel Flight Ticket

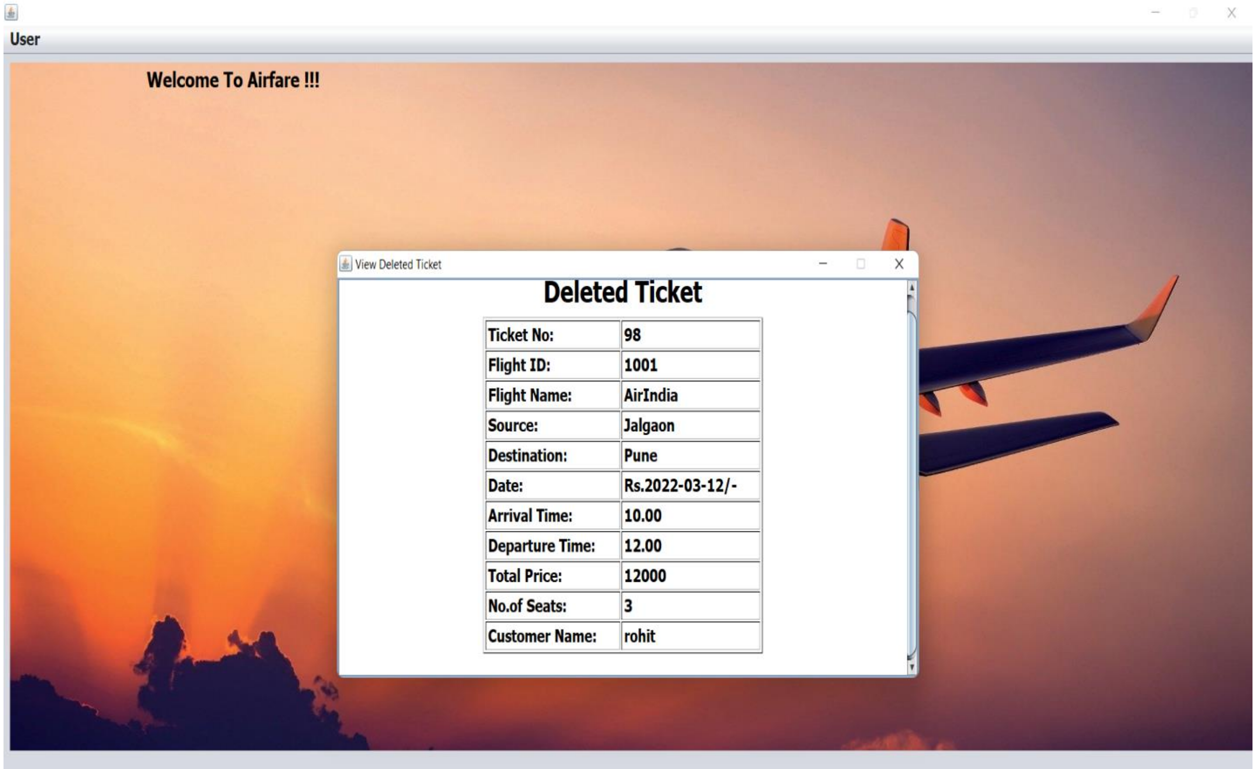
Ticket No:	98	Arrival Time:	10.00
Flight Id:	1001	Departure Time:	12.00
Flight Name:	AirIndia	Price:	12000
Source:	Jalgaon	Seats:	3
Destination:	Pune	Customer Name:	rohit
Booking Date:	12 Mar, 2022	Cancel date:	14 Mar, 2022

Message

Ticket Cancel

Print deleted ticket :

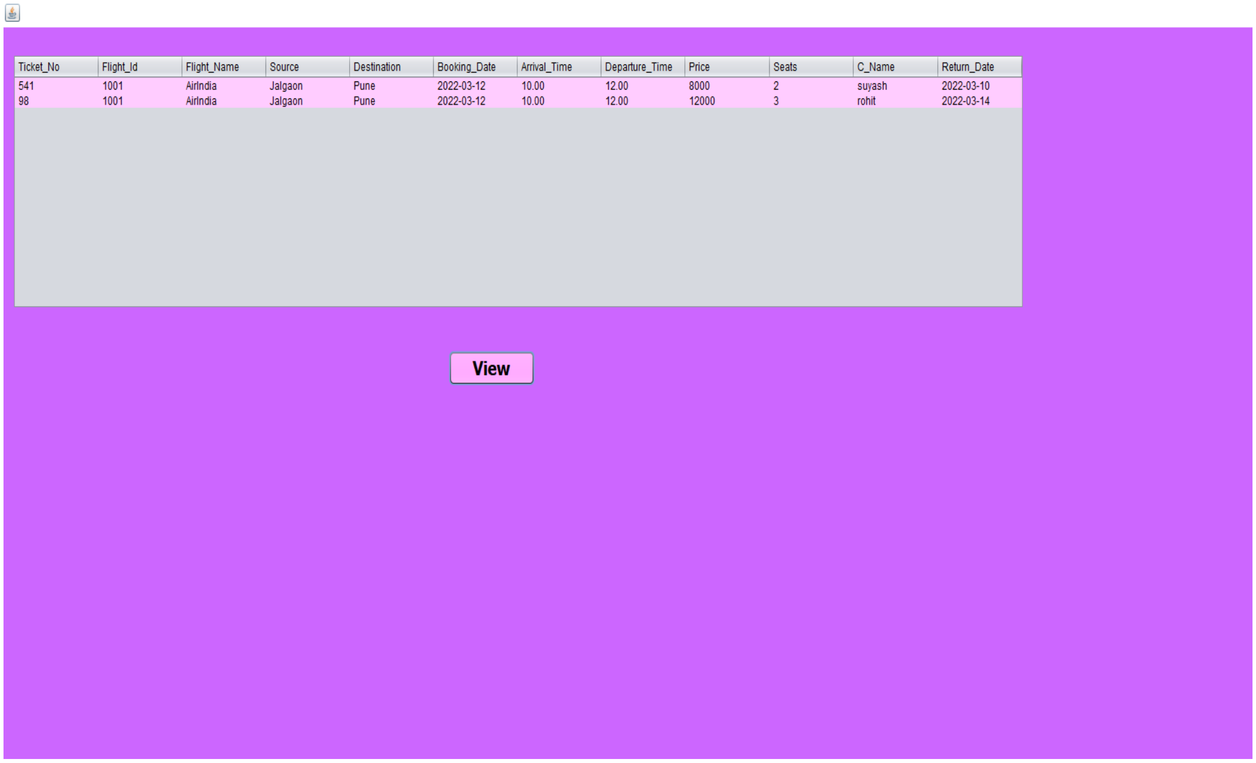




Deleted Ticket

Ticket No:	98
Flight ID:	1001
Flight Name:	AirIndia
Source:	Jalgaon
Destination:	Pune
Date:	Rs.2022-03-12/-
Arrival Time:	10.00
Departure Time:	12.00
Total Price:	12000
No.of Seats:	3
Customer Name:	rohit

View Deleted tickets:



Ticket_No	Flight_Id	Flight_Name	Source	Destination	Booking_Date	Arrival_Time	Departure_Time	Price	Seats	C_Name	Return_Date
541	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	8000	2	suyash	2022-03-10
98	1001	AirIndia	Jalgaon	Pune	2022-03-12	10.00	12.00	12000	3	rohit	2022-03-14

3.6 Table specifications

1. **Admin :**

Entity Name	Data Type	Size	Description
Username	varchar	50	Admin-name for login
Password	varchar	50	Password for login

2. **Account :**

Name	Data Type	Size	Description
Username	varchar	100	User-name to login
Name	varchar	100	Actual name of user
Password	varchar	100	Password for the account
Sec_Q	varchar	100	Security question. Like (what is your nick name)
Answer	varchar	250	Answer of the question.

3. Flight :

Name	Data Type	Size	Description
Flight_Id	int(primary key)	100	To give flight id
Flight_Name	varchar	100	To give flight name
Source	varchar	100	To start the journey from which place
Destination	varchar	100	to end the journey at which place.
Date	varchar	100	To give the date
Arrival_Time	varchar	50	Arriving time
Departure_Time	varchar	50	Departure time
Flight_Price	int	100	Price of particular flight.

4. Book Ticket :

Name	**Data Type**	**Size**	**Description**
Ticket_No	int(primary key)	50	Ticket number
Flight_Id	int	50	Flight Id
Flight_Name	varchar	50	Name of flight
source	varchar	50	Starting place of journey
Destination	varchar	50	Ending place of journey.
Date	varchar	50	To give the date
Arrival_Time	varchar	50	Arriving time
Departure_Time	varchar	50	Departure time
Total_Price	int	50	Price of flight
Seats	int	50	Number of seats
C_Name	varchar	100	To give customer name

5. **Cancel Ticket:**

Name	Data Type	Size	Description
Ticket_No	int	50	Ticket number
Flight_Id	int	50	Flight Id
Flight_Name	varchar	100	Name of the flight
source	varchar	100	Starting place of journey
Destination	varchar	100	Ending place of journey
Booking Date	varchar	50	To give the date
Arrival_Time	varchar	50	Arriving time
Departure_Time	varchar	50	Departure time
Price	int	100	Price of flight
Seats	int	40	No. of seats
C_Name	varchar	200	Customer name
Return_Date	varchar	50	To give return date

3.7 Data Dictionary:

Data Dictionary is a store of information about the data in database. The dictionary defines the name, description, source of data, users of data, and keywords in data, formula to derive the data, specification and such other details. Data dictionary brings common understanding of the data in the organization. RDBMS provides software to create the dictionary. Use of data dictionary enforces the standards of processing, usage's application and documentation in the organization.

Data Dictionaries are an integral component of structured analysis, since data flow diagram by them do not fully describe the information about the system. The data dictionary provides additional information about the system.

A data dictionary is a catalogue – a repository – of the elements in a system. These elements center on data the way they are structured to meet user requirements and organization needs. In a data dictionary, a list of all the elements composing the data flowing through a system is included. If a project team member wants to know the definition of a data item name or the contents of a particular data flow, the information will be available in the data dictionary. Descriptions of all data used in the system are given in a data dictionary.

3.8 Test Procedures and Implementation

Test procedure

The software testing is the critical element of software quality assurance and represents the ultimate review of the software design and coding. The main objective of the testing is to find an error and to uncover the errors that are not yet discovered.

The increasing visibility of software as a system element and the attendant cost associated with a software failure and motivating forces for well planned, through testing. It is no unusual for a software development organization to expand between 30% to 40% of project effort on testing. In the extreme, testing of human related software can cost 3-5 time as much as all other software engineering activities combined. the testing phase involves the testing of the system using various test data, preparation of the test data plays a vital role in the system testing after preparing the test data, error where found and corrected by using the following the testing steps and correction are recorded for future reference. Thus a series of testing is performed on the system before it is ready for implementation.

After completion of system analysis, design and coding through testing of the system was carried out in a systematic approach, the main objectives of the system are

- To ensure that the operations of the system will perform as per the specification.
- To make sure that the system meets the user requirement during the operations.
- To cross check the when correct input are filled into the system output are correct.
- To make sure that during the operation incorrect inputs and the outputs will be detected.

In testing process the number of strategies have been used as mentioned below,

- Unit Testing
- Integration Testing
- Validation Testing
- Black Box Testing
- User acceptance Testing

Unit Testing

Unit testing focuses verification efforts on the smallest unit of the software design. Using the system test plan, prepare in the design phase of the system development as guide, important control path are tested to uncover error within boundary of the module. The interface of each of the module was tested to ensure proper flow of information into and out of the module under consideration. Each module will be tested individually so as to make the individual component error free. Also other attached modules will be error free.

Integration Testing:

Each module will be tested of its effect on other module by integrating the modules. This will remove further errors from the system and may also result in some changes in the individual module.

Validation Testing

At the culmination of the integration testing the software was completely assembled as package, interfaces have been uncovered, and a final series of software validation testing began. Here we test the system function manner that can be reasonably by the customer ,the system was tested against system requirement specification.

Black Box Testing:

After performing validation testing, the next phase is output test of the system, since no system code is useful if it does not produce the desired output in desired format. By considering the format of the report/output, report/output is generated or displayed and tested.

User Acceptance Testing:

User acceptance testing is used to determine the whether the software is fit for the user to use. The System under consideration was listed for user acceptance by keeping constant touch with the prospective user of the system at the time of design, development and making change whenever required.

4: USER MANUAL

4.1 User Manual

Although the user interface of the system is constructed in such a way that anyone can use the system if he has the basic knowledge of the operating keyboard and mouse operation of the computer. All pages of the application contain the descriptive links and the buttons that will help the user to perform the required operation.

There are following links/module.

Admin:

Admin will have the full authority of the software. Admin will login by using his account. Admin will view/Edit the Flight details. Admin will be able to delete the flight details .

Registration :

User has to register himself/herself through signup option in AirFare .Non-registered user cannot login to our project.

Forget Password :

Any registered user while login forgot his/her password, then through forget password ,user can retrieve the password over here. And login to airfare successfully.

Login :

To login, registered user has to give username and password then the user is able to access Add Flights, Book Tickets, Cancel Tickets ,Veiw Booked Ticket, View Cancelled Tickets.

Book Ticket:

Here user can book ticket by giving ticket_no, source, destination, date, price, seat and customer name.

Payment Option:

Here after booking ticket user has to do payment by any of the one option Debit card / Credit card. And then the ticket is booked successfully.

Print Ticket:

Onces the ticket is booked. User can see the print of ticket here. By just giving ticket-no. entire ticket is displayed.

Cancel Ticket:

Here any user is willing to cancel the booked ticket. Then the user has to give only ticket_no and click search. Entire details will be displayed and just to click on cancel.

View Ticket:

Here user can see the entire details of booked tickets, just after clicking on view.

View deleted ticket:

Here user can see the entire details of cancellation of booked tickets. That is deleted tickets, After clicking on view.

4.2 Reports

The “REPORT” contain information about the Book tickets, Cancel tickets and Flight details, etc. In this report there is all information regarding filghts

In Book Ticket report ticket no, flight Id, flight Name, source, destination, Date, arrival and departure time, total price seats name is there

In booking cancel report ticket no, flight Id, flight Name, source, destination, Booking Date, arrival and departure time, price, seats name is there

In flight Report Flight Id, flight name, source, destination, date, arrival and departure time, flight price is there.

5 : Drawbacks and limitations

1) Excel report has not been developed for Airline Ticket, Cancelled Tickets due to some criticality.
2) In this project, customers cannot postpone their flights.
3) Inability of customers to select their preferred seats.
4) Customers capture and modification is not possible.

6 : Proposed Enhancements

- Current system is designed in short amount of time so all functionality are not included in the system.
- We might add Facility to book a ticket with Hotels too, while booking a ticket for Airline.
- We may allow customers to postpone their flights and even customers to select their preferred seats.
- Implement the backup mechanism for taking backup of codebase and database on regular basis on different servers.

7 : Conclusion

The Airfare system has been a way of minimizing the clerical work, which is almost a routine and consumes the most precious time.

This Airfare system has been an attempt to help the user to minimize his workload along with minimizing the paper works and saving of time.

The system has been developed in a way to make it very user friendly. It provides an on-line message and an error detection and error messages every time the user needs. Any person having a little bit of window based can run this system without any pain.

Almost all the difficulties of manual reservation have been removed by this system. Ti wind up let me welcome all the suggestions and other improvements, which the system needs so that it covers all the needs if the user in the user way.

8 : References:

Websites:

1. www.roseindia.net
2. www.google.co.in
3. www.wikipedia.com
4. www.tutorialspoint
5. https://www.tutorialspoint.com/java/
6. https://www.javatpoint.com/java-tutorial
7. https://www.tutorialspoint.com/mysql/

9 SAMPLE PROGRAM CODE:

login.java

```
import java.awt.Color;
import java.sql.Connection;
import java.sql.PreparedStatement;
import java.sql.ResultSet;
import javax.swing.JOptionPane;
import javax.swing.UIManager;
public class Login extends javax.swing.JFrame {
    Connection conn;
    ResultSet rs;
    PreparedStatement pst;

    /**
     * Creates new form Login
     */
    public Login() {
        super("Login");
        initComponents();
        conn=javaconnect.ConnectDb();
        Color col=new Color(255,255,255);
        getContentPane().setBackground(col);

    }


    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
    private void initComponents() {

        jPanel6 = new javax.swing.JPanel();
        jLabel1 = new javax.swing.JLabel();
        jLabel2 = new javax.swing.JLabel();
        jTextField1 = new javax.swing.JTextField();
        jPasswordField1 = new javax.swing.JPasswordField();
        Admin = new javax.swing.JButton();
        Login = new javax.swing.JButton();
        Signup = new javax.swing.JButton();
        Forgot = new javax.swing.JButton();
        jLabel3 = new javax.swing.JLabel();
        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);

        setBackground(new java.awt.Color(255, 255, 255));
        setResizable(false);
        getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jLabel3.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/Logo for
airfare.png"))); // NOI18N
        getContentPane().add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(650, 150,
410, 250));

        jPanel6.setBackground(new java.awt.Color(204, 153, 255));
        jPanel6.setBorder(javax.swing.BorderFactory.createEmptyBorder(1, 1, 1, 1));
        jPanel6.setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());

        jLabel1.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        jLabel1.setText("Username");
        jPanel6.add(jLabel1, new org.netbeans.lib.awtextra.AbsoluteConstraints(20, 35, 120, 29));

        jLabel2.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        jLabel2.setText("Password");
        jPanel6.add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(14, 91, 120, 26));
        jPanel6.add(jTextField1, new org.netbeans.lib.awtextra.AbsoluteConstraints(158, 36, 184, 29));
        jPanel6.add(jPasswordField1, new org.netbeans.lib.awtextra.AbsoluteConstraints(158, 91, 184,
29));

        Admin.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        Admin.setText("Admin");
        Admin.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                new Admin();
                dispose();
            }
        });
        jPanel6.add(Admin, new org.netbeans.lib.awtextra.AbsoluteConstraints(14, 148, -1, -1));

        Login.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        Login.setText("Login");
        Login.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                LoginActionPerformed(evt);
            }
        });
        jPanel6.add(Login, new org.netbeans.lib.awtextra.AbsoluteConstraints(110, 148, -1, -1));

        Signup.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        Signup.setText("Signup");
        Signup.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                SignupActionPerformed(evt);
            }
        });
        jPanel6.add(Signup, new org.netbeans.lib.awtextra.AbsoluteConstraints(200, 148, -1, -1));

        Forgot.setFont(new java.awt.Font("Tahoma", 1, 20)); // NOI18N
        Forgot.setText("Forgot Password");
        Forgot.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                ForgotActionPerformed(evt);
            }
        });
        jPanel6.add(Forgot, new org.netbeans.lib.awtextra.AbsoluteConstraints(300, 148, -1, -1));

        getContentPane().add(jPanel6, new org.netbeans.lib.awtextra.AbsoluteConstraints(100, 150,
550, 270));

        pack();
        setExtendedState(MAXIMIZED_BOTH);
    }// </editor-fold>//GEN-END:initComponents

    private void LoginActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_LoginActionPerformed
        // TODO add your handling code here:
        String sql="select * from Account where Username=? and Password=?";
        try{
            pst=conn.prepareStatement (sql);

            pst.setString(1,jTextField1.getText());
            pst.setString(2,jPasswordField1.getText());
            rs=pst.executeQuery();
            if(rs.next()){
                rs.close();
                pst.close();

                setVisible(false);
                // Loading ob=new Loading();
                //ob.setUpLoading();
                UserHome ob=new UserHome();
                ob.setVisible(true);

            }else{
                JOptionPane.showMessageDialog(null, "Login Failed");
            }

        }catch(Exception e){
            JOptionPane.showMessageDialog(null, e);
        }finally{
            try{
                rs.close();
                pst.close();
            }catch(Exception e){
            }
        }

    }//GEN-LAST:event_LoginActionPerformed

    private void SignupActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_SignupActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Signup ob=new Signup();
        ob.setVisible(true);
    }//GEN-LAST:event_SignupActionPerformed

    private void ForgotActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_ForgotActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Forgot ob=new Forgot();
        ob.setVisible(true);
    }//GEN-LAST:event_ForgotActionPerformed


    public static void main(String args[]) {

        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (IllegalAccessException ex) {
```

```
java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Login.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Login().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton Forgot;
    private javax.swing.JButton Login;
    private javax.swing.JButton Signup;
    private javax.swing.JButton Admin;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JPanel jPanel6;
    private javax.swing.JPasswordField jPasswordField1;
    private javax.swing.JTextField jTextField1;
// End of variables declaration//GEN-END:variables
```

Home.java

```
import javax.swing.*;
import java.awt.*;
import java.awt.event.*;


public class Home extends javax.swing.JFrame {

    /**
     * Creates new form Home
     */
    public Home() {
        initComponents();
    }


    @SuppressWarnings("unchecked")
    // <editor-fold defaultstate="collapsed" desc="Generated Code">//GEN-BEGIN:initComponents
    private void initComponents() {


        mbMain = new JMenuBar();

        mnuAdmin = new JMenu("Admin");

        miAddFlight = new JMenuItem("Add Flight");
        miBackToLogin = new JMenuItem("Back to Login");
        miExit = new JMenuItem("Exit");

        mnuAdmin.add(miAddFlight);
        mnuAdmin.add(miBackToLogin);
        mnuAdmin.addSeparator();
        mnuAdmin.add(miExit);

        mbMain.add(mnuAdmin);

        miAddFlight.setFont(new java.awt.Font("Tahoma", 1, 20));
        miBackToLogin.setFont(new java.awt.Font("Tahoma", 1, 20));
        miExit.setFont(new java.awt.Font("Tahoma", 1, 20));
        mnuAdmin.setFont(new java.awt.Font("Tahoma", 1, 20));

        jLabel1 = new javax.swing.JLabel();
        Add_Flight = new javax.swing.JButton();
        jButton5 = new javax.swing.JButton();
        Ticket_Cancel = new javax.swing.JButton();
        exit = new javax.swing.JButton();
        search_flight = new javax.swing.JButton();
        back_to_login = new javax.swing.JButton();
        jButton1 = new javax.swing.JButton();
        jButton2 = new javax.swing.JButton();
        jLabel2 = new javax.swing.JLabel();
        jLabel3 = new javax.swing.JLabel();

        jLabel1.setIcon(new javax.swing.ImageIcon(getClass().getResource("/Images/airp (1).jpg"))); //
  NOI18N

        setDefaultCloseOperation(javax.swing.WindowConstants.EXIT_ON_CLOSE);
        setAlwaysOnTop(true);
        setBackground(new java.awt.Color(204, 204, 204));
        setCursor(new java.awt.Cursor(java.awt.Cursor.DEFAULT_CURSOR));
        setPreferredSize(new java.awt.Dimension(1000, 500));
        setResizable(false);
        setJMenuBar(mbMain);
        getContentPane().setLayout(new org.netbeans.lib.awtextra.AbsoluteLayout());
/*
        Add_Flight.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        Add_Flight.setText("Add Flight");
        Add_Flight.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                Add_FlightActionPerformed(evt);
            }
        });
        getContentPane().add(Add_Flight, new org.netbeans.lib.awtextra.AbsoluteConstraints(87, 102,
-1, 32));

        jButton5.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        jButton5.setText("Book Ticket");
        jButton5.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton5ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton5, new org.netbeans.lib.awtextra.AbsoluteConstraints(347, 102, -
1, 32));

        Ticket_Cancel.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        Ticket_Cancel.setText("Ticket Cancel");
        Ticket_Cancel.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                Ticket_CancelActionPerformed(evt);
            }
        });
        getContentPane().add(Ticket_Cancel, new org.netbeans.lib.awtextra.AbsoluteConstraints(564,
102, -1, 32));

        exit.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        exit.setText("Exit");
        exit.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                exitActionPerformed(evt);
            }
        });
        getContentPane().add(exit, new org.netbeans.lib.awtextra.AbsoluteConstraints(590, 166, -1,
32));

        search_flight.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        search_flight.setText("Search Flight Details");
        search_flight.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                search_flightActionPerformed(evt);
            }
        });
        getContentPane().add(search_flight, new org.netbeans.lib.awtextra.AbsoluteConstraints(318,
166, -1, 32));

        back_to_login.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        back_to_login.setText("Back to login page");
        back_to_login.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                back_to_loginActionPerformed(evt);
            }
        });
        getContentPane().add(back_to_login, new org.netbeans.lib.awtextra.AbsoluteConstraints(77,
166, -1, 32));

        jButton1.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        jButton1.setText("View Ticket");
        jButton1.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton1ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton1, new org.netbeans.lib.awtextra.AbsoluteConstraints(92, 244,
120, -1));

        jButton2.setFont(new java.awt.Font("Tahoma", 1, 14)); // NOI18N
        jButton2.setText("View Deleted Ticket");
        jButton2.addActionListener(new java.awt.event.ActionListener() {
            public void actionPerformed(java.awt.event.ActionEvent evt) {
                jButton2ActionPerformed(evt);
            }
        });
        getContentPane().add(jButton2, new org.netbeans.lib.awtextra.AbsoluteConstraints(318, 244,
190, -1));
*/
        jLabel2.setFont(new java.awt.Font("Tahoma", 1, 24)); // NOI18N
        jLabel2.setText("Welcome To Airfare !!!");
        getContentPane().add(jLabel2, new org.netbeans.lib.awtextra.AbsoluteConstraints(220, 13, 411,
32));

        jLabel3.setIcon(new
javax.swing.ImageIcon(getClass().getResource("/Images/background.jpg"))); // NOI18N
        jLabel3.setPreferredSize(new java.awt.Dimension(2000, 800));
        getContentPane().add(jLabel3, new org.netbeans.lib.awtextra.AbsoluteConstraints(10, 10, 2500,
800));

        pack();
```

```
                setExtendedState(JFrame.MAXIMIZED_BOTH);

                miExit.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent ae) {
                                System.exit(0);
                        }
                });

                miAddFlight.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent ae) {
                                Add_FlightActionPerformed(ae);
                        }
                });

                miBackToLogin.addActionListener(new ActionListener() {
                        public void actionPerformed(ActionEvent ae) {
                                back_to_loginActionPerformed(ae);
                        }
                });

    }// </editor-fold>//GEN-END:initComponents

    private void Add_FlightActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_Add_FlightActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Add_Flight_Details ob=new Add_Flight_Details();
        ob.setVisible(true);
    }//GEN-LAST:event_Add_FlightActionPerformed

    private void jButton5ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButton5ActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Search_Flight_Details ob=new Search_Flight_Details();
        ob.setVisible(true);
    }//GEN-LAST:event_jButton5ActionPerformed

    private void Ticket_CancelActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_Ticket_CancelActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Cancel_Flight_Ticket ob=new Cancel_Flight_Ticket();
        ob.setVisible(true);
    }//GEN-LAST:event_Ticket_CancelActionPerformed

    private void back_to_loginActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_back_to_loginActionPerformed
```

```
        // TODO add your handling code here:
        setVisible(false);
        Login ob=new Login();
        ob.setVisible(true);
    }//GEN-LAST:event_back_to_loginActionPerformed

    private void search_flightActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_search_flightActionPerformed
        // TODO add your handling code here:
        setVisible(false);
        Search_Flight_Details ob=new Search_Flight_Details();
        ob.setVisible(true);
    }//GEN-LAST:event_search_flightActionPerformed

    private void exitActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_exitActionPerformed
        // TODO add your handling code here:
        JFrame frame=new JFrame("EXIT");
        if(JOptionPane.showConfirmDialog(frame, "Confirm if you want to exit","EXIT",
                JOptionPane.YES_NO_OPTION)==JOptionPane.YES_NO_OPTION)
        {
            System.exit(0);
        }
    }//GEN-LAST:event_exitActionPerformed

    private void jButton1ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButton1ActionPerformed

new viewtickets().setVisible(true);


// TODO add your handling code here:
    }//GEN-LAST:event_jButton1ActionPerformed

    private void jButton2ActionPerformed(java.awt.event.ActionEvent evt) {//GEN-
FIRST:event_jButton2ActionPerformed

        new Deletedtickets().setVisible(true);


   // TODO add your handling code here:
    }//GEN-LAST:event_jButton2ActionPerformed


    public static void main(String args[]) {

        try {
            for (javax.swing.UIManager.LookAndFeelInfo info :
```

```
javax.swing.UIManager.getInstalledLookAndFeels()) {
                if ("Nimbus".equals(info.getName())) {
                    javax.swing.UIManager.setLookAndFeel(info.getClassName());
                    break;
                }
            }
        } catch (ClassNotFoundException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (InstantiationException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (IllegalAccessException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        } catch (javax.swing.UnsupportedLookAndFeelException ex) {

java.util.logging.Logger.getLogger(Home.class.getName()).log(java.util.logging.Level.SEVERE,
null, ex);
        }
        //</editor-fold>

        /* Create and display the form */
        java.awt.EventQueue.invokeLater(new Runnable() {
            public void run() {
                new Home().setVisible(true);
            }
        });
    }

    // Variables declaration - do not modify//GEN-BEGIN:variables
    private javax.swing.JButton Add_Flight;
    private javax.swing.JButton Ticket_Cancel;
    private javax.swing.JButton back_to_login;
    private javax.swing.JButton exit;
    private javax.swing.JButton jButton1;
    private javax.swing.JButton jButton2;
    private javax.swing.JButton jButton5;
    private javax.swing.JLabel jLabel1;
    private javax.swing.JLabel jLabel2;
    private javax.swing.JLabel jLabel3;
    private javax.swing.JButton search_flight;
    // End of variables declaration//GEN-END:variables
    private JMenuBar mbMain;
    private JMenuItem miAddFlight;
    private JMenuItem miBackToLogin;
    private JMenuItem miExit;
    private JMenu mnuAdmin;
}
```

** Thankyou**

****Have nice day and enjoy your flight****